HARDWARE REFERENCE MANUAL

Turbo PMAC Clipper

(Turbo PMAC2-Eth-Lite)

Turbo PMAC2-Eth-Lite Hardware Reference

4xx-603871-xAxx

December 6, 2010

DELTA TAU
Data Systems, Inc.
NEW IDEAS IN MOTION ...

Single Source Machine Control *Power // Flexibility // Ease of Use*
21314 Lassen Street Chatsworth, CA 91311 // Tel. (818) 998-2095 Fax. (818) 998-7807 // www.deltatau.com

Copyright Information

To report errors or inconsistencies, call or email:

Delta Tau Data Systems, Inc. Technical Support
Phone: (818) 717-5656
Fax: (818) 998-7807
Email: support@deltatau.com
Website: http://www.deltatau.com

Operating Conditions

All Delta Tau Data Systems, Inc. motion controller products, accessories, and amplifiers contain static sensitive components that can be damaged by incorrect handling. When installing or handling Delta Tau Data Systems, Inc. products, avoid contact with highly insulated materials. Only qualified personnel should be allowed to handle this equipment.

In the case of industrial applications, we expect our products to be protected from hazardous or conductive materials and/or environments that could cause harm to the controller by damaging components or causing electrical shorts. When our products are used in an industrial environment, install them into an industrial electrical cabinet or industrial PC to protect them from excessive or corrosive moisture, abnormal ambient temperatures, and conductive materials. If Delta Tau Data Systems, Inc. products are exposed to hazardous or conductive materials and/or environments, we cannot guarantee their operation.

REVISION HISTORY				
REV.	**DESCRIPTION**	**DATE**	**CH G**	**APPVD**
1	NEW MANUAL CREATION	03/23/07	CP	S. MILICI
2	CORRECTIONS TO JUMPERS E4, E5, E6	04/25/07	CP	S. MILICI
3	UPGRADE FROM PRELIMINARY STATUS; ENET IP SETUP PP. 12-16; ADD CH5 OPT12 MOTOR P. 24	11/13/07	CP	S. MILICI
4	UPDATED CPU ANALOG INPUTS, P. 23	11/22/07	CP	S. SATTARI
5	UPDATES FOR VERSION –103 AND –104	05/06/08	CP	S. MILICI
6	REVISED +5V POWER SUPPLY SPECS, P. 6 18 AWG WIRE REQ. FOR +5V AND GND (PP. 6 & 45) NEW DIMENSIONED BOARD LAYOUT DRAWING CORRECTED J4 TYPOS FOR PINS 25-26 (P. 40) CORRECTED DIGITAL I/O CHANNEL NAMES (PP. 25-26) J9 PINOUT – NOTES FOR E16, E17 & 10k PULLUP (P. 42)	10/23/08	CP	C. COKER
7	UPDATED DESC. OF E1 & E2, E5, E8 (P.5) ADDED E8 JUMPER NOTE RE: IP ADDRESS (PPS. 12 & 14) UPDATED JUMPERS E5 & E8 FOR CLARITY (P. 34)	11/19/08	CP	C. COKER
8	UPDATED SETUP OF 5TH MOTOR USING OPT 12, P. 26 ADDED OPT-11A APPLICATION NOTE, P. 30	10/15/09	CP	S. MILICI
9	CHANGED NAME OF MANUAL TO TURBO PMAC CLIPPER	11/03/09	CP	D. DIMITRI
10	ADJUSTED DIAGRAM ON P.31	12/16/09	CP	S. MILICI
11	Added pulse and direction setup pg 26. Updated fifth motor setup pg 35.	6/10/2010	RN	S.MILICI

Table of Contents

INTRODUCTION

The Turbo PMAC2-Eth-Lite controller ("Clipper") from Delta Tau provides a very powerful, but compact and cost-effective, multi-axis controller for cost-sensitive applications. It has a full Turbo PMAC2 CPU section and provides a minimum of 4 axes of servo or stepper control with 32 general-purpose digital I/O points. It provides both Ethernet and RS-232 communications links.

The optional axis expansion board provides a set of four additional servo channels and extra I/O ports.

Board Configuration

Base Version

- The base version of the Clipper Controller (Turbo PMAC2-Eth-Lite) provides a 110mm x 220mm (4.25" x 8.5") board with:
- 80 MHz DSP56303 Turbo PMAC CPU
- 256k x 24 user SRAM
- 1M x 8 flash memory for user backup & firmware
- Latest released firmware version
- RS-232 serial interface
- 100 Mbps Ethernet interface
- 480 Mbit/sec USB 2.0 interface
- 4 channels axis-interface circuitry, each including:
  - 12-bit +10V analog output
  - Pulse-&-direction digital outputs
  - 3-channel differential/single-ended encoder input
  - 5 input flags, 2 output flags
  - UVW TTL-level "hall" inputs
- 50-pin IDC header for amplifier/encoder interface
- 34-pin IDC header for flag interface
- 4-pin Molex connector for power supply input (5V, +/-12V, GND)
- (+/-12V only required for analog outputs or inputs)
- PID/notch/feedforward servo algorithms
- 32 general-purpose TTL-level I/O points, direction selectable by byte:
  - 16-point multiplexer port compatible with Delta Tau I/O accessories
  - 16-point "Opto" port compatible with Opto-22-style modules
- "Handwheel" port with 2 each:
  - Quadrature encoder inputs
  - Pulse (PFM or PWM) output pairs

On-board options:

- Optional 2 channels 12-bit A/D converters, 1 12-bit D/A converter
- Optional Modbus Ethernet I/O protocol
- On-board 8K x 16 dual-ported RAM.

Stackable accessories supported:

- ACC-1P PC/104-format Channel 5-8 board
- ACC-8ES 4-channel dual 18-bit true-DAC output board
- ACC-8FS 4-channel direct-PWM output board

- ACC-8TS 4-channel ADC-interface board
- ACC-51S 2/4-channel high-resolution encoder interpolator board

Board Options

Option 5xx: CPU Speed Options

- OPT-5C3 80MHz DSP56303 CPU, expanded program and user data memory
- OPT-5F3 240MHz DSP56321 CPU, expanded program memory and user data memory

Option 10: Firmware Version Specification

Normally the Turbo PMAC2-Eth-Lite Controller is provided with the newest released firmware version. A label on the memory IC shows the firmware version loaded at the factory. Option 10 provides for a user-specified firmware version.

Option 12: Analog-to-Digital Converters

Option 12 permits the installation of two channels of on-board analog-to-digital converters with ±10V input range and 12-bits resolution. This option also provides one filtered PWM DAC output.

Additional Accessories

Acc-1P: Axis Expansion Piggyback Board

Acc-1P provides four additional channels axis interface circuitry for a total of eight servo channels, each including:

- 12-bit ±10V analog output
- Pulse-and-direction digital outputs
- 3-channel differential/single-ended encoder input
- Four input flags, two output flags
- Three PWM top-and-bottom pairs (unbuffered)

Acc-1P Option 1: I/O Ports

Option 1 provides the following ports on the Acc-1P axes expansion board for digital I/O connections.

- Multiplexer Port: This connector provides eight input lines and eight output lines at TTL levels. When using the PMAC Acc-34x type boards these lines allow multiplexing large numbers of inputs and outputs on the port. Up to 32 of the multiplexed I/O boards may be daisy-chained on the port, in any combination.
- I/O Port: This port provides eight general-purpose digital inputs and eight general-purpose digital outputs at 5 to 24Vdc levels. This 34-pin connector was designed for easy interface to OPTO-22 or equivalent optically isolated I/O modules when different voltage levels or opto-isolation to the PMAC2A PC/104 is necessary.
- Handwheel port: this port provides two extra channels, each jumper selectable between encoder input or pulse output.

Acc-1P Option 2: Analog-to-Digital Converters

Option 2 permits the installation on the Acc-1P of two channels of analog-to-digital converters with ±10V input range and 12-bits resolution.

Acc-8TS Connections Board

Acc-8TS is a stack interface board to for the connection of either one or two Acc-28B A/D converter boards. When a digital amplifier with current feedback is used, the analog inputs provided by the Acc-28B cannot be used.

Acc-8ES Four-Channel Dual-DAC Analog Stack Board

Acc-8ES provides four channels of 18-bit dual-DAC with four DB-9 connectors. This accessory is stacked to the Clipper Board and it is mostly used with amplifiers that require two ±10 V command signals for sinusoidal commutation.

Acc-8FS Four-Channel Direct PWM Stack Breakout Board

Acc-8FS is a 4-channel direct PWM stack breakout board for the Clipper Board. This is used for controlling digital amplifiers that require direct PWM control signals. When a digital amplifier with current feedback is used, the analog inputs provided by the Option 12 of the Clipper Board (the Option 2 of the Acc-1P or the Acc-28B) cannot be used.

Acc-51S Four-Channel High Resolution Interpolator Board

The Acc-51S Interpolator Accessory is a sine wave input interpolator designed to interface analog quadrature encoders to the Clipper Board. The Acc-51S stacks on top of the Clipper Board or on top of the Acc-1P 5-8 axis board. The Interpolator accepts inputs from two (optionally four) sinusoidal or quasi-sinusoidal encoders and provides encoder position data to the PMAC. This interpolator creates 4,096 steps per sine-wave cycle.

HARDWARE SETUP

On the Clipper Board, there are a number of jumpers called E-points or W-points that customize the hardware features of the CPU for a given application and must be setup appropriately. The following is an overview grouped in appropriate categories. For an itemized description of the jumper setup configuration, refer to the E-Point Descriptions section.

Configuration Jumpers

E0: Forced Reset Control Jumper – Remove E0 for normal operation. Installing E0 forces PMAC to a reset state. This configuration is for factory use only; the board will not operate with E0 installed.

E1 and E2: Serial Port Selection Jumper (rev 102 and lower only) – These jumpers select the target CPU for the serial port as either the main PMAC CPU or the Ethernet CPU (change IP address). Both jumpers must be set the same.

- 1-2 for Main CPU
- 2-3 for Ethernet CPU

E3: Re-Initialization on Reset Control Jumper – If E3 is OFF (default), PMAC executes a normal reset, loading active memory from the last saved configuration in non-volatile flash memory. If E3 is ON, PMAC re-initializes on reset, loading active memory with the factory default values.

E4: Watchdog Timer Disable Jumper – Jumper E4 must be OFF for the watchdog timer to operate. This is a very important safety feature, so it is vital that this jumper be OFF for normal operation. E4 should only be put ON to debug problems with the watchdog timer circuit.

E5: For factory use only:

Rev 102 and higher - Jumper E5 must be removed during normal operation

Rev 101 and lower - E5 must be installed on pins 1 to 2 during normal operation

E6: ADC Enable Jumper – Install E6 to enable the analog-to-digital converter circuitry ordered through Option-12. Remove this jumper to disable this option, which might be necessary to control motor 1 through a digital amplifier with current feedback.

E8: USB/EtherNet Write Protect Jumper – Remove E8 prior to changing the IP address

E10-E12: Power-Up State Jumpers – Jumper E10 must be OFF, jumper E11 must be ON, and jumper E12 must be ON, in order for the CPU to copy the firmware from flash memory into active RAM on power-up/reset. This is necessary for normal operation of the card. (Other settings are for factory use only.)

E13: Firmware Load Jumper – If jumper E13 is ON during power-up/reset, the board comes up in bootstrap mode which permits loading of firmware into the flash-memory IC. When the PMAC Executive program tries to establish communications with a board in this mode, it will detect automatically that the board is in bootstrap mode and ask what file to download as the new firmware. Jumper E13 must be OFF during power-up/reset for the board to come up in normal operational mode.

E14-E17: Ports Direction Control Jumpers – These jumpers select the I/O lines direction of the JTHW and the JOPT connectors. This allows configuring these ports as all inputs, all outputs or half inputs and half outputs according to the following tables:

JTHW Connector			
E14	**E15**	**DATx lines**	**SELx lines**
OFF	OFF	Output	Output
OFF	ON	Output	Input
ON	OFF	Input	Output
ON	ON	Input	Input

JOPT Connector			
E16	**E17**	**MOx lines**	**MIx Lines**
OFF	OFF	Output	Output
OFF	ON	Output	Input
ON	OFF	Input	Output
ON	ON	Input	Input

If E14 is removed or E15 is installed then the multiplexing feature if the JTHW port cannot be used.

MACHINE CONNECTIONS

Typically, the user connections are made to terminal blocks that attach to the JMACH connectors by a flat cable. The following are the terminal blocks recommended for connections:

- 34-Pin IDC header to terminal block breakouts (Phoenix part number 2281063) Delta Tau part number 100-FLKM34-000
- 50-Pin IDC header to terminal block breakouts (Phoenix part number 2281089) Delta Tau part number 100-FLKM50-000

Mounting

The Clipper Board is typically installed as a stand-alone controller using standoffs. At each of the four corners of the board and at the center edges, there are mounting holes that can be used for this.

The order of the Acc-1P or other stacked accessories with respect to the Clipper Board does not matter.

Power Supplies

Digital Power Supply

6A @ +5V (±5%) (15 W) with a minimum 5 msec rise time

(Eight-channel configuration, with a typical load of encoders)

The Clipper Board and other stackable accessories each require a 1A @ 5VDC power supply for normal operation, however the Clipper board has an "in-rush" current requirement that can reach 3A so a 6A @ 5VDC power supply is recommended. The +5V and ground reference lines from the power supply should be connected to TB1 terminal block of the Clipper board using 18 AWG stranded wire.

DAC Outputs Power Supply

0.3A @ +12 to +15V (4.5W)
0.25A @ -12 to -15V (3.8W)
(Eight-channel configuration)

The ±12V lines from the supply, including the ground reference, can be brought in either from the TB1 terminal block or from the JMACH1 connector.

Flags Power Supply

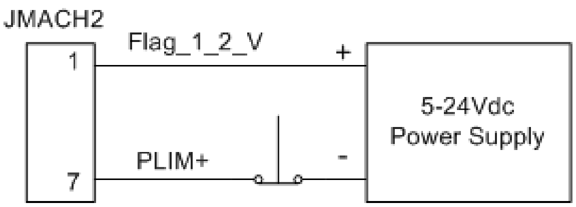
Each channel of PMAC has five dedicated digital inputs on the machine connector: PLIMn, MLIMn (overtravel limits), HOMEn (home flag), FAULTn (amplifier fault), and USERn. A power supply from 5 to 24V must be used to power the circuits related to these inputs. This power supply can be the same used to power PMAC and can be connected from the TB1 terminal block or the JMACH1 connector.

Overtravel Limits and Home Switches

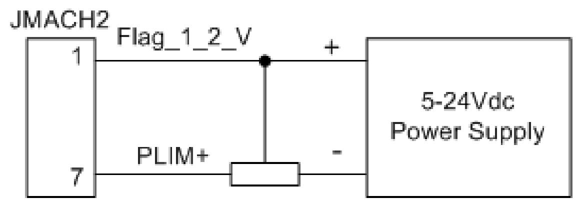
When assigned for the dedicated uses, these signals provide important safety and accuracy functions. PLIMn and MLIMn are direction-sensitive over-travel limits that must conduct current to permit motion in that direction. If no over-travel switches will be connected to a particular motor, this feature must be disabled in the software setup through the PMAC Ixx24 variable.

Types of Overtravel Limits

PMAC expects a closed-to-ground connection for the limits to not be considered on fault. This arrangement provides a failsafe condition. Usually, a passive normally close switch is used. If a proximity switch is needed instead, use a 5 to 24V normally closed to ground NPN sinking type sensor.



Dry Contact



15-24V proximity

Home Switches

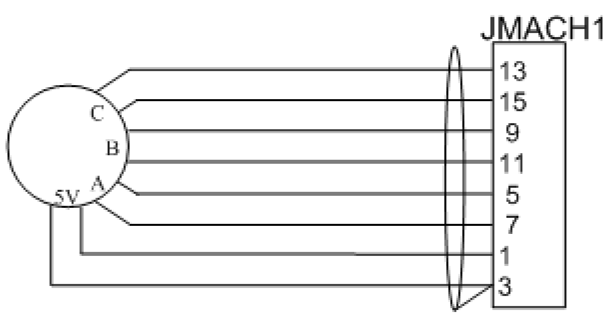
While normally closed-to-ground switches are required for the overtravel limits inputs, the home switches could be either normally close or normally open types. The polarity is determined by the home sequence setup, through the I-variables I9n2.

Motor Signals Connections

Incremental Encoder Connection

Each JMACH1 connector provides two +5V outputs and two logic grounds for powering encoders and other devices. The +5V outputs are on pins 1 and 2; the grounds are on pins 3 and 4. The encoder signal pins are grouped by number: all those numbered 1 (CHA1+, CHA1-, CHB1+, CHC1+, etc.) belong to encoder #1. The encoder number does not have to match the motor number, but usually does. Connect the A and B (quadrature) encoder channels to the appropriate terminal block pins. For encoder 1, the CHA1+ is pin 5 and CHB1+ is pin 9. If there is a single-ended signal, leave the complementary signal pins floating – do not ground them. However, if single-ended encoders are used, check the setting of the resistor packs (see the Hardware Setup section for details). For a differential encoder, connect the complementary signal lines – CHA1- is pin 7, and CHB1- is pin 11. The third channel (index pulse) is optional; for encoder 1, CHC1+ is pin 13, and CHC1- is pin 15.

Example: differential quadrature encoder connected to channel #1:



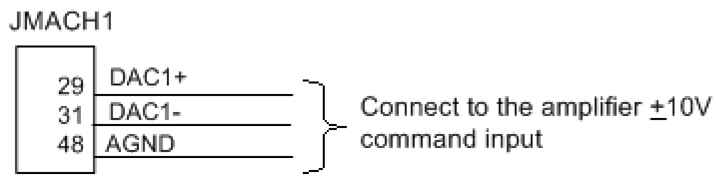
DAC Output Signals

If PMAC is not performing the commutation for the motor, only one analog output channel is required to command the motor. This output channel can be either single-ended or differential, depending on what the amplifier is expecting. For a single-ended command using PMAC channel 1, connect DAC1+ (pin 29) to the command input on the amplifier. Connect the amplifier's command signal return line to PMAC's GND line (pin 48). In this setup, leave the DAC1- pin floating; do not ground it.

For a differential command using PMAC channel 1, connect DAC1 (pin 29) to the plus-command input on the amplifier. Connect DAC1- (pin 31) to the minus-command input on the amplifier. PMAC's GND should still be connected to the amplifier common.

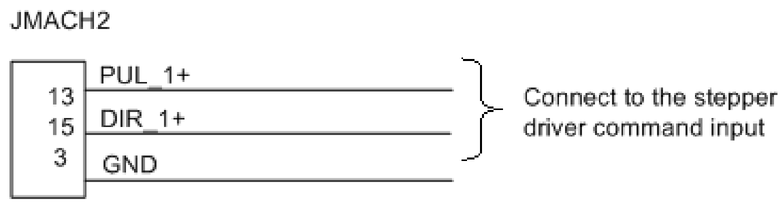
Any analog output not used for dedicated servo purposes may be utilized as a general-purpose analog output by defining an M-variable to the command register, then writing values to the M-variable. The analog outputs are intended to drive high-impedance inputs with no significant current draw (10mA max). The 220Ω output resistors will keep the current draw lower than 50 mA in all cases and prevent damage to the output circuitry, but any current draw above 10 mA can result in noticeable signal distortion.

Example:



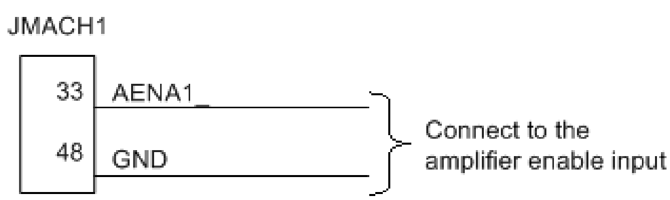
Pulse and Direction (Stepper) Drivers

The channels provided by the Clipper Board or the Acc-1P board can output pulse and direction signals for controlling stepper drivers or hybrid amplifiers. These signals are at TTL levels.



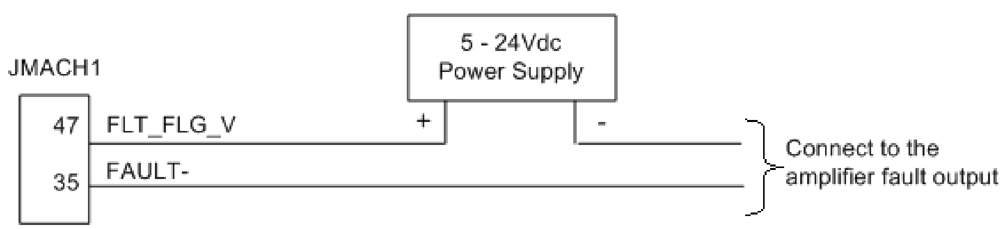
Amplifier Enable Signal (AENAn/DIRn)

Most amplifiers have an enable/disable input that permits complete shutdown of the amplifier regardless of the voltage of the command signal. PMAC's AENA line is meant for this purpose. AENA1- is pin 33. This signal is an open-collector output and an external 3.3 kΩ pull-up resistor can be used if necessary.



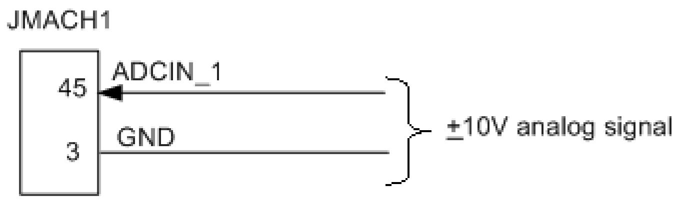
Amplifier Fault Signal (FAULT-)

This input can take a signal from the amplifier so PMAC knows when the amplifier is having problems, and can shut down action. The polarity is programmable with I-variable Ixx24 (I124 for motor 1) and the return signal is ground (GND). FAULT1- is pin 35. With the default setup, this signal must actively be pulled low for a fault condition. In this setup, if nothing is wired into this input, PMAC will consider the motor not to be in a fault condition.



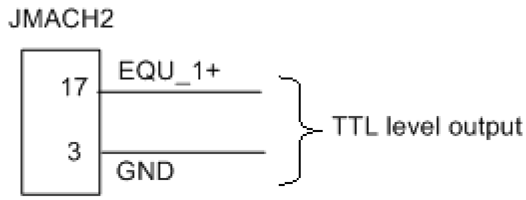
Optional Analog Inputs

The optional analog-to-digital converter inputs are ordered either through Option-12 on the CPU or Option-2 on the axis expansion board. Each option provides two 12-bit analog inputs analog inputs with a ±10Vdc range, and one 12-bit filtered PWM DAC output.



Compare Equal Outputs

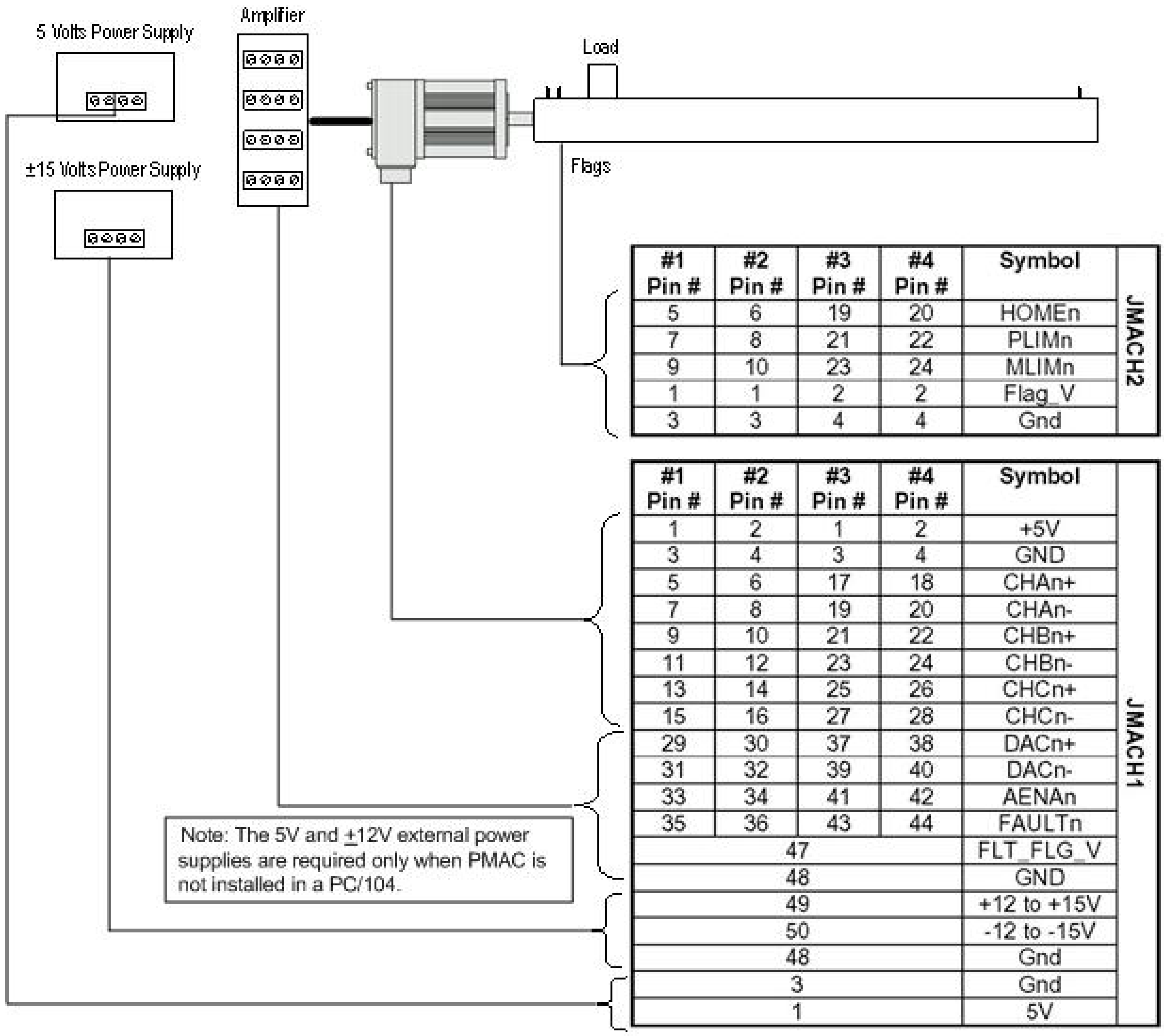
The compare-equals (EQU) outputs have a dedicated use of providing a signal edge when an encoder position reaches a pre-loaded value. This is very useful for scanning and measurement applications. Instructions for use of these outputs are covered in detail in the PMAC2 User Manual.



Serial Port (JRS232 Port)

For serial communications, use a serial cable to connect your PC's COM port to the J2 serial port connector present on the Clipper Board. Delta Tau provides the Acc-3L cable for this purpose that connects the PMAC to a DB-9 connector. Standard DB-9-to-DB-25 or DB-25-to-DB-9 adapters may be needed for your particular setup.

Machine Connections Example: Using Analog ±10V Amplifier



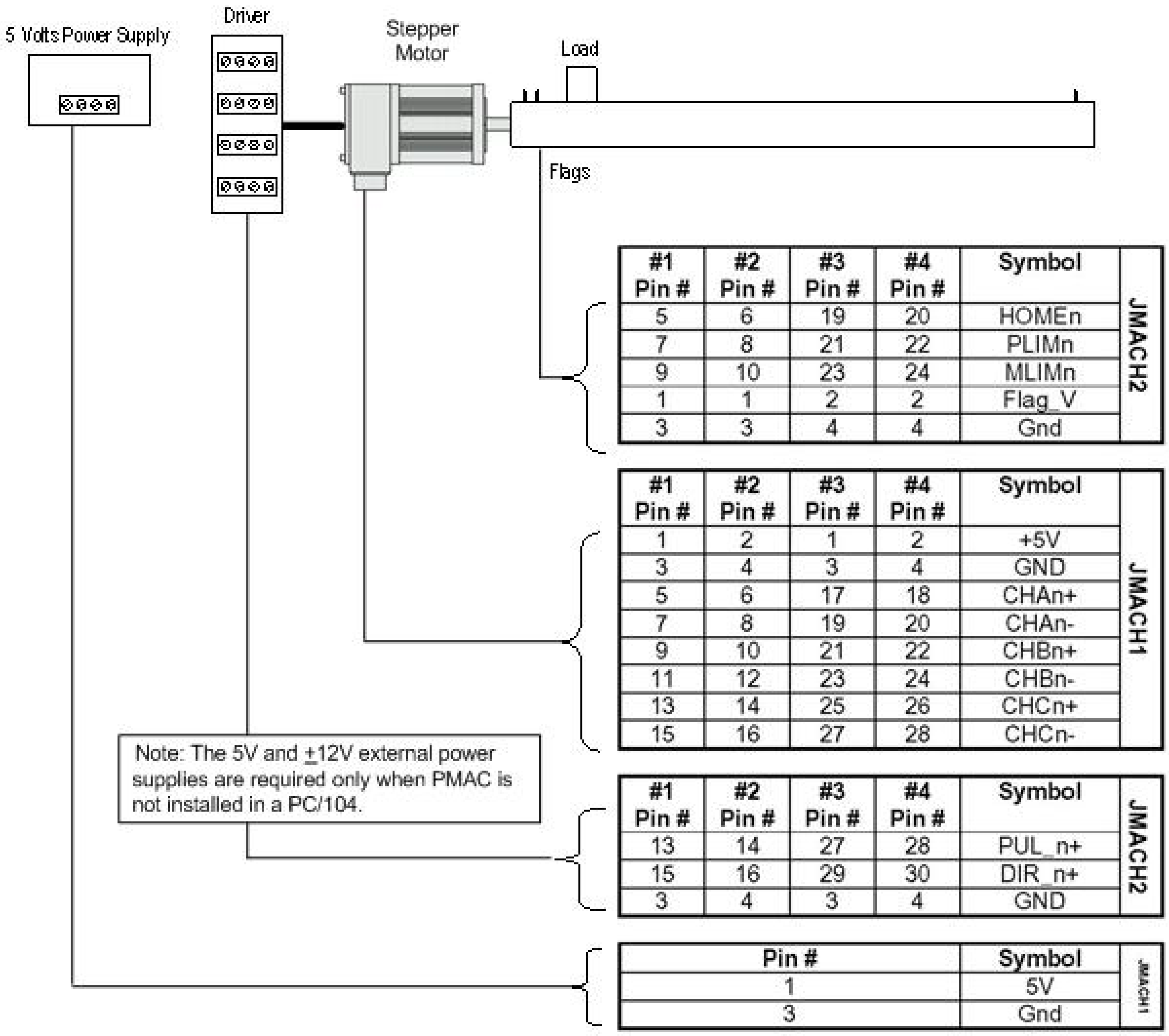
JMACH2

#1 Pin #	#2 Pin #	#3 Pin #	#4 Pin #	Symbol
5	6	19	20	HOMEn
7	8	21	22	PLIMn
9	10	23	24	MLIMn
1	1	2	2	Flag_V
3	3	4	4	Gnd

JMACH1

#1 Pin #	#2 Pin #	#3 Pin #	#4 Pin #	Symbol
1	2	1	2	+5V
3	4	3	4	GND
5	6	17	18	CHAn+
7	8	19	20	CHAn-
9	10	21	22	CHBn+
11	12	23	24	CHBn-
13	14	25	26	CHCn+
15	16	27	28	CHCn-
29	30	37	38	DACn+
31	32	39	40	DACn-
33	34	41	42	AENAn
35	36	43	44	FAULTn
47				FLT_FLG_V
48				GND
49				+12 to +15V
50				-12 to -15V
48				Gnd
3				Gnd
1				5V

Machine Connections Example: Using Pulse and Direction Drivers



#1 Pin #	#2 Pin #	#3 Pin #	#4 Pin #	Symbol	
5	6	19	20	HOMEn	JMACH2
7	8	21	22	PLIMn	
9	10	23	24	MLIMn	
1	1	2	2	Flag_V	
3	3	4	4	Gnd	

#1 Pin #	#2 Pin #	#3 Pin #	#4 Pin #	Symbol	
1	2	1	2	+5V	JMACH1
3	4	3	4	GND	
5	6	17	18	CHAn+	
7	8	19	20	CHAn-	
9	10	21	22	CHBn+	
11	12	23	24	CHBn-	
13	14	25	26	CHCn+	
15	16	27	28	CHCn-	

#1 Pin #	#2 Pin #	#3 Pin #	#4 Pin #	Symbol	
13	14	27	28	PUL_n+	JMACH2
15	16	29	30	DIR_n+	
3	4	3	4	GND	

Pin #	Symbol	
1	5V	JMACH1
3	Gnd	

SOFTWARE SETUP

PMAC I-Variables

PMAC has a large set of Initialization parameters (I-variables) that determine the "personality" of the card for a specific application. Many of these are used to configure a motor properly. Once set up, these variables may be stored in non-volatile EAROM memory (using the **SAVE** command) so the card is always configured properly (PMAC loads the EAROM I-variable values into RAM on power-up).

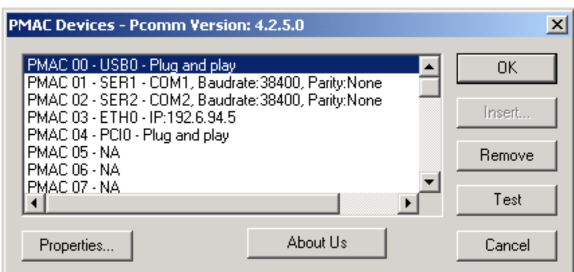
The programming features and configuration variables for the Clipper Board are described fully in the Turbo PMAC User and Software manuals.

Communications

Delta Tau provides software tools that allow communication with the Clipper Board via its standard RS-232 port, USB or Ethernet ports. The PEWIN32 Pro2 Executive is the most important in the series of software accessories, and it allows configuring and programming the PMAC for any particular application.

Configuring IP address through the Ethernet port using PeWin32 Pro2

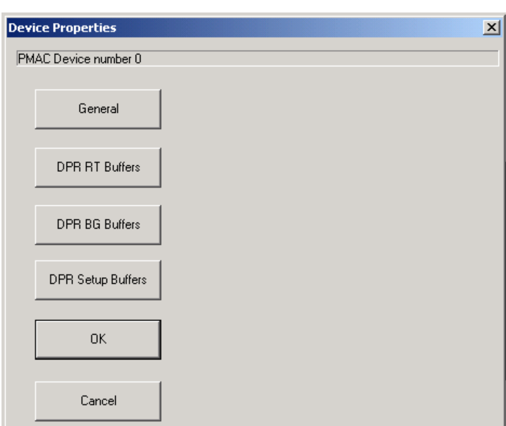
In the PMAC Devices window select the PMAC Ethernet device that you wish to change (as in PMAC 03 below) and click on the "Properties…" button:



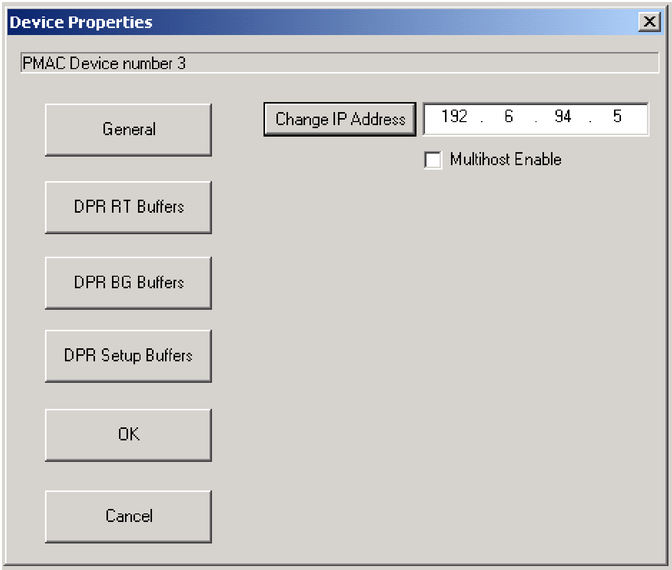
Note:

Jumper E8 must be installed to enable changing of the IP address.

Click the **General** button in the **Device Properties** window:



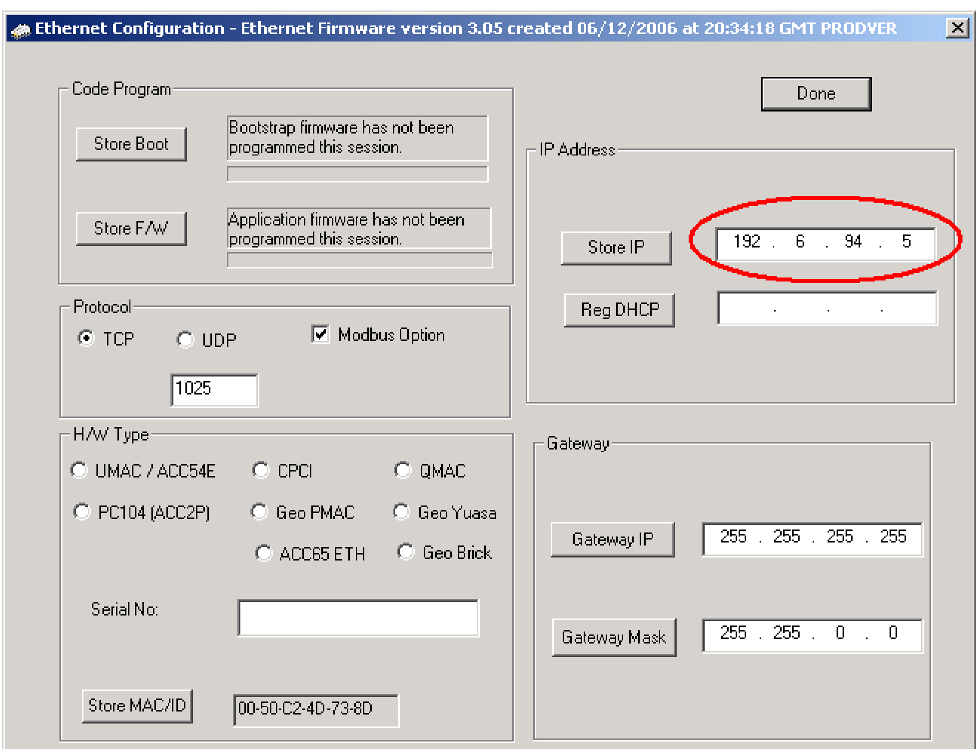
When this window appears, click the **Change IP Address** button and set the new address:



It will take effect on the next power cycle. You must now change the address in the **PMAC Devices** window of the Pro2 Executive.

Configuring the IP address with the "EthUSBConfigure.exe" application.

Connect the USB cable and power on the PMAC. Launch the application: "EthUSBConfigure.exe".



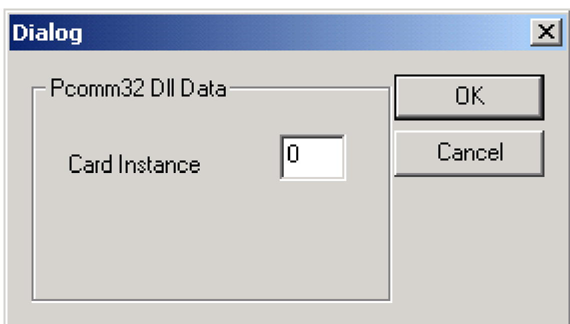
Enter the new IP address in the box shown above.

Note:

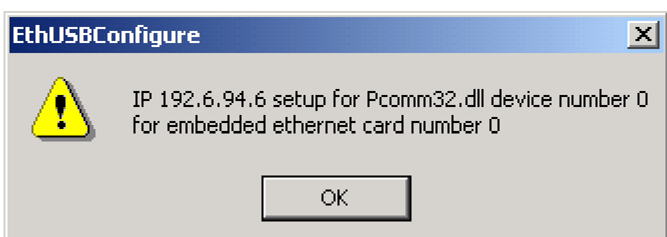
Jumper E8 must be installed to enable changing of the IP address.

Click the **Store IP** button. When the following dialog appears, click **Yes**.

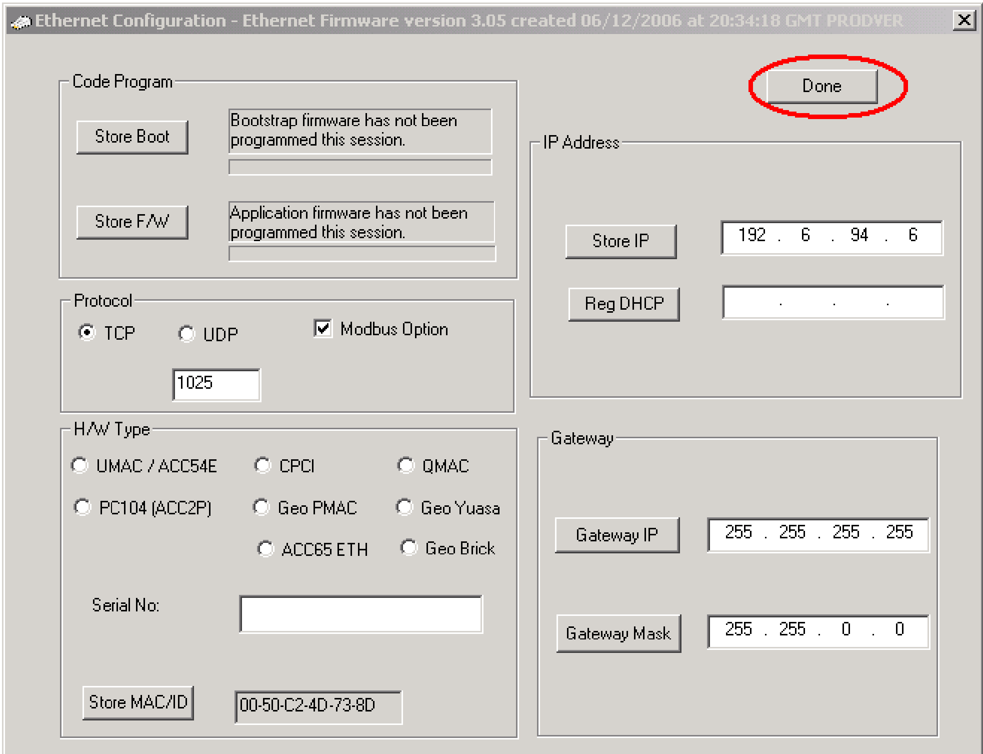
The next dialog will appear. If this is the only instance of an IP address, leave the Card Instance value at zero. If you have multiple instances of IP addresses (multiple PMAC EtherNet cards), enter the instance in the box and click **OK**.



When the following dialog appears, click **OK**.



Click the **Done** button. This will take effect on the next PMAC power cycle. You must now change the address in the **PMAC Devices** window of your PMAC application.



Configuring USB

Starting with Pewin Pro and Service Pack 2.0, the USB driver support for this revision of the card is bundled with the Pewin Pro installation program. The UMAC USB card will work only with Windows 98, Windows ME, Windows 2000 and Windows XP. It will not function with Windows NT 4.0; this version of Windows does not support plug and play, which is required by all USB devices.

Note:

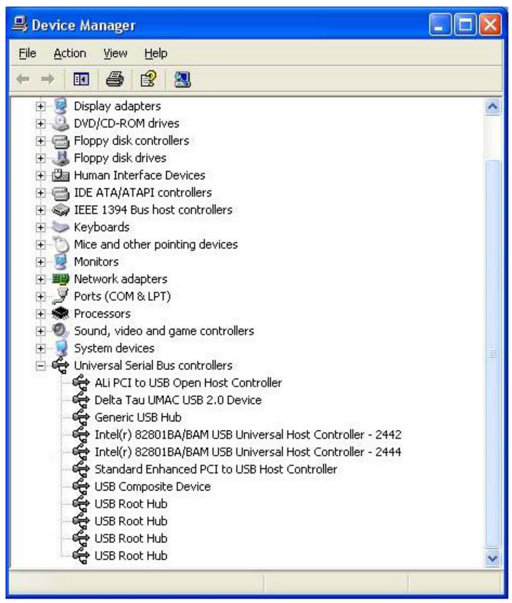
Windows XP is recommended since the UMAC has on-board USB 2.0 and only Windows XP has native USB 2.0 support.

One file is placed on the PC to achieve USB connectivity – device driver PMACUSB.SYS in the WINDOWS\SYSTEM32\DRIVERS directory and the PMACUSB.INF plug and play information file in the WINDOWS\INF directory. When the UMAC is plugged into the PC, a **New Hardware Found** message displays. A series of dialog boxes will appear, indicating that Windows is installing the device drivers for the system.

Note:

Plug in the USB cable from the UMAC to the PC after the software Pewin Pro and its Service Pack 2.0 has been installed. If the USB cable is plugged in before the software has been installed, restart Windows.

To verify that the software device drivers have been installed properly, right-click on the **My Computer** icon on the desktop. Select **Properties** from the drop-down menu that appears. The **System Properties Windows** dialog box appears. Click the tab titled **Device Manager**. At this point, a list of device categories appears. Click the + to see a list of USB devices. Provided the device driver for the UMAC Turbo CPU/ Communications Board has been installed properly, a dialog box displays, similar to the following:



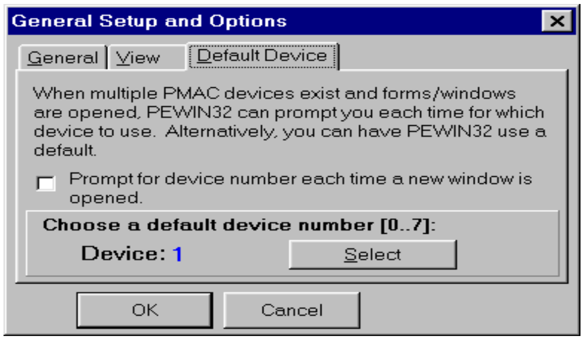
If Delta Tau UMAC USB 2.0 Device is not on the list, the device driver has not been installed. If there is a red x through that line or a yellow exclamation point through that line, then Windows had a problem installing the device.

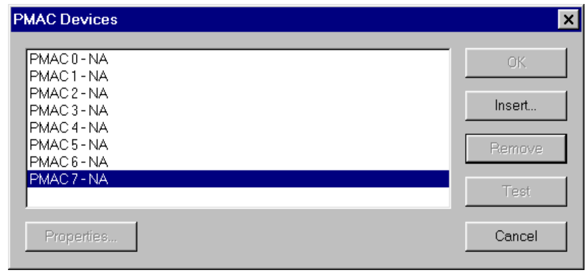
The appropriate trouble-shooting steps are:

- Reboot the computer and examine this list again.
- If that does not work, ensure that pmacusb.sys is in the Windows\system32\Drivers directory.
- If this is true, when using an older computer, check with the manufacturer to make sure that there is not an update to the BIOS to enable USB on the PC.
- If the Universal Serial Bus Controllers in the device manager dialog box are not on the list, make sure that it is enabled in the BIOS of the computer.

Using PeWin32 Pro and later to establish communication

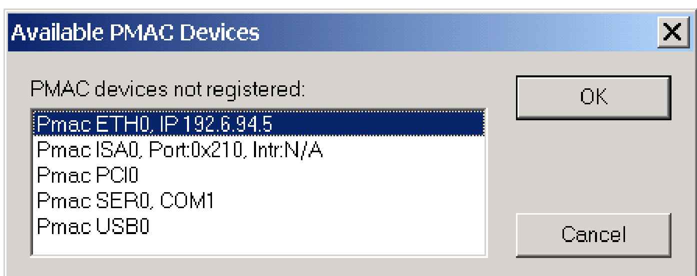
Once the driver is installed, it needs additional configuration by using the PmacSelect dialog. The PmacSelect dialog is accessible by all programs created with PComm 32 Pro (via the PmacSelect() function call). Launch the supplied Delta Tau application (Pewin 32 Pro, PMAC Test Pro, or any application) from the program menu and display the **PmacSelect** dialog.





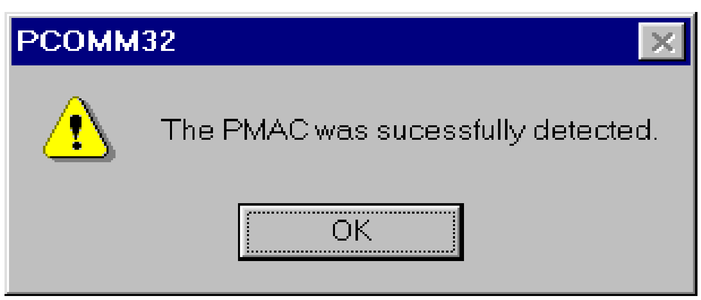
Product	Display the PmacSelect Dialog
Pewin 32 Pro	From the main menu item setup, go to Setup\General Setup and Options. Select the **Default Device** tab. Click on the **Select** button.
Pcomm 32 Pro	Run the supplied PmacTest application. From the main menu, select **Configure\Communications**. Also, the **`PmacSelect()`** function can be called from any application that has been coded.
Ptalk DT Pro	Call the **`SelectDevice()`** method of Ptalk from the supplied or self-created programs.

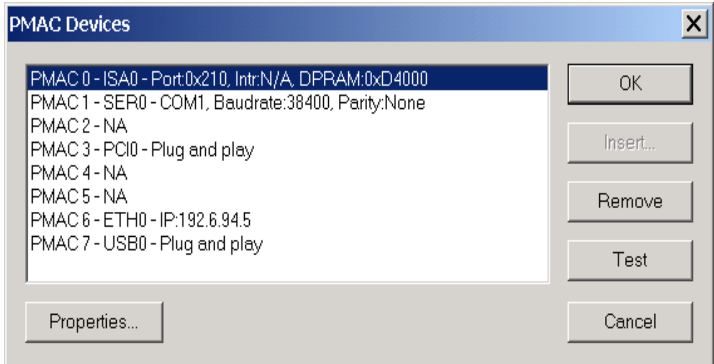
From the device selection screen, select the device number to insert a device and click Insert. Another window listing all configured devices will appear.



Select the device to configure and click **OK**.

Once a PMAC is listed in the Pmacselect window, it is registered and can accept communication. It is recommended to test a device upon registering. At this time, the following screen displays and this device is ready for use in any application.




PMAC Devices
PMAC 0 - ISA0 - Port:0x210, Intr:N/A, DPRAM:0xD4000
PMAC 1 - SER0 - COM1, Baudrate:38400, Parity:None
PMAC 2 - NA
PMAC 3 - PCI0 - Plug and play
PMAC 4 - NA
PMAC 5 - NA
PMAC 6 - ETH0 - IP:192.6.94.5
PMAC 7 - USB0 - Plug and play
OK
Insert...
Remove
Test
Properties...
Cancel

Operational Frequency and Baud Rate Setup

I52 controls the operational clock frequency of the Turbo PMAC's CPU by controlling the multiplication factor of the phase-locked loop (PLL) inside the CPU. The PLL circuit multiplies the input 10 MHz (actually 9.83 MHz) clock frequency by a factor of (I52 + 1) to create the clock frequency for the CPU. Formally, this is expressed in the equation:

CPU Frequency (MHz) = 10 * (I52 + 1)

I52 should usually be set to create the highest CPU frequency for which the CPU is rated. For the standard 80 MHz CPU, it should be set to 7.

Note:

It may be possible to operate a CPU at a frequency higher than its rated frequency, particularly at low ambient temperatures. However, safe operation cannot be guaranteed under these conditions, and any such operation is done entirely at the user's own risk.

I52 is actually used at power-on/reset only, so to make a change in the CPU frequency with I52, change the value of I52, store this new value to non-volatile flash memory with the SAVE command, and reset the card with the $$$ command.

If too high a value of I52 has been set, the watchdog timer on the PMAC will likely trip immediately after reset due to CPU operational failure. If this happens, the PMAC must be reinitialized using E3.

Filtered DAC Output Configuration

The Clipper Board +/-10V DAC outputs are produced by filtering a PWM signal. Although this technique does not contain the same levels of performance as a true Digital to Analog converter, for most servo applications it is more than adequate. This section is meant for explaining the tradeoffs of PWM frequency vs. resolution in the Clipper Board configuration as well as a comparison to the true 18 bit DACs.

Both the resolution and the frequency of the Filtered PWM outputs are configured in software on the Clipper Board through the variable **I7000**. This variable also effects the phase and servo interrupts. Therefore as we change **I7000** we will also have to change **I7001** (phase clock divider), **I7002** (servo clock divider), and **I10** (servo interrupt time). These four variables are all related and must be understood before adjusting parameters.

Note:

Note that the Filtered DAC Output Configuration sets the Max PWM frequency very high (29KHz). This setting can be problematic with Direct PWM commutation on the same servo IC. Also the ACC-28A and ACC-28B can not be used on the same servo IC since the PWM frequency settings are out of range for these products.

Since the Clipper Board uses standard Turbo PMAC2 firmware the following I-variables must be set properly to use the digital-to-analog (filtered DAC) outputs:

```
I7000  = 1001              ; PWM frequency 29.4kHz, PWM 1-4
I7001  = 5                 ; Phase Clock 9.8kHz
I7002  = 3                 ; Servo frequency 2.45kHz
I7003  = 1746              ; ADC frequency
I7100  = 1001              ; PWM frequency 29.4kHz, PWM 5-8
I7103  = 1746              ; ADC frequency
I70n6  = 0                 ; Output mode: PWM
Ixx69  = 1001              ; DAC limit 10Vdc
I10    = 3421867           ; Servo interrupt time
```

n = channel number from 1 to 8
xx = motor number from 1 to 8

Parameters to Set up Global Hardware Signals

I7000

I7000 determines the frequency of the **MaxPhase** clock signal from which the actual phase clock signal is derived. It also determines the PWM cycle frequency for Channels 1 to 4. This variable is set according to the equation:

```
I7000 = INT[117,964.8/(4*PWMFreq(KHz)) - 1]
```

The Clipper Board filtered PWM circuits were optimized for about 30KHz. The minimum frequency **I7000** should be set to is 1088 (calculated as 27.06856KHz)

I7001

I7001 determines how the actual phase clock is generated from the **MaxPhase** clock, using the equation:

```
PhaseFreq(kHz) = MaxPhaseFreq(kHz)/(I7001+1)
```

I7001 is an integer value with a range of 0 to 15, permitting a division range of 1 to 16. Typically, the phase clock frequency is in the range of 8 kHz to 12 kHz. About 9 KHz is standard, set **I7001** = 5.

I7002

I7002 determines how the servo clock is generated from the phase clock, using the equation:

```
ServoFreq(KHz) = PhaseFreq(KHz)/(I7002+1)
```

I7002 is an integer value with a range of 0 to 15, permitting a division range of 1 to 16. On the servo update, which occurs once per servo clock cycle, PMAC updates commanded position (interpolates) and closes the position/velocity servo loop for all active motors, whether or not commutation and/or a digital current loop is closed. Typical servo clock frequencies are 1 to 4 kHz. The PMAC standard is about 2 KHz, set **I902** = 3.

I10 tells the Clipper Board interpolation routines how much time there is between servo clock cycles. It must be changed any time **I7000**, **I7001**, or **I7002** is changed. **I10** can be set according to the formula:

```
I10 = (2*I7000+3)*(I7001+1)*(I7002+1)*640/9
```

I10 should be set to `3421867`.

I7003

I7003 determines the frequency of four hardware clock signals used for machine interface channels 1-4; This can be left at the default value (I7003=*) unless the on board Option-12 ADCs are used. The four hardware clock signals are SCLK (encoder sample clock), PFM_CLK (pulse frequency modulator clock), DAC_CLK (digital-to-analog converter clock), and ADC_CLK (analog-to-digital converter clock).

Parameters to Set Up Per-Channel Hardware Signals

I70n6

I70n6 is the output mode; “n” is the output channel number (i.e. for channel 1 the variable to set would be I7016, I7026 for channel 2 etc.). On Pmac1, there is only one output and one output mode: DAC output. On PMAC2 boards, each channel has 3 outputs, and there are 4 output modes. Since this board was designed to output filtered PWM signals, we want to configure at least the first output as PWM. Therefore the default value of 0 is the choice. For information on this variable consult the Turbo Software Reference Manual.

Ixx69

Ixx69 is the motor output command limit. The analog outputs on PMAC1 style boards and some PMAC2 accessories are 16-bit or 18-bit DACs, which map a numerical range of -32,768 to +32,767 into a voltage range of -10V to +10V relative to analog ground (AGND). For our purposes of a filtered PWM output this value still represents the maximum voltage output; however, the ratio is slightly different. With a true DAC, Ixx69=32767 allows a maximum voltage of 10V output. With the filtered PWM circuit, Ixx69 is a function of I7000. A 10V signal in the output register is no longer 32767 as was in PMAC1, a 10V signal is corresponds to a value equal to I7000. Anything over I7000 will just rail the DAC at 10V. For example:

Desired Maximum Output Value = 6V

```
Ixx69 = 6/10 * I7000
```

Desired Maximum Output Value = 10V

```
Ixx69= I7000 + 10  ; add a little headroom to assure a full
10V
```

Effects of Changing I7000 on the System

It should now be understood that a full 10 volts is output when the output register is equal to I7000. The output register is suggested m-variable Mxx02 (I.e. M102->Y:$078002,8,16,S ; OUT1A command value; DAC or PWM). With default setting of I7000, 10 volts is output when M102 is equal to I7000, or 1001. Since the output register is an integer value the smallest increment of change is about 10mV (1/1001 * 10V). Some users may want to calibrate their analog output using Ixx29. Ixx29 is an integer similar to Mxx02 except the value is added to the output register every servo cycle to apply a digital offset to the output register. Therefore the resolution of our output signal affects how Ixx29 should be set. As mentioned earlier, with the default parameters, 1 bit change in the output register changes the analog output by about 10mV. Therefore if there is an analog output offset less than 5mV, Ixx29 cannot decrease your offset. By increasing I7000 you increase your resolution, so if you double I7000, 1 bit change in the output register corresponds to about 5mV. So with Ixx29 you can only change the offset in increments of 5mV.

You can see above that by increasing I7000 you increase the resolution of our command output register. While this does offer some advantages, users should carefully consider the tradeoffs when changing I7000 between resolution and ripple.

By increasing I7000 we are essentially decreasing our PWM Frequency. The two are related by the following equation:

```
I7000 = INT[117,964.8/(4*PWMFreq(KHz)) - 1]
```

Passing the PWM signal through a 10KHz low pass filter creates the +/-10V signal output. The duty cycle of the PWM signal is what generates the magnitude the voltage output. The frequency of the PWM signal determines the magnitude and frequency of ripple on that +/-10V signal. As you lower the PWM frequency and subsequently increase your output resolution, you increase the magnitude of the ripple as well as slow down the frequency of the ripple as well. Depending on the system, this ripple can affect performance at different levels.

So what do we mean by ripple? Ripple is the small signal that will you will see on top of the +/-10V signal if you put an oscilloscope on it. In other words, if users command a 4V signal out of the Clipper Board and scope it, the result is a small sinusoidal type wave centered on 4V. At the default PWM frequency and output resolution this will have a magnitude of about 250mV to 450mV and a frequency of about 30kHz. This is typically faster than any of the control loops so the amplifier essentially filters it out of the system.

For example, to double the resolution of the output signal, users merely double the I7000 value from 1001 to 2002. How does this affect the ripple? Testing shows the ripple magnitude to increase from around 300mV to well over 800mV. The frequency of the ripple decreased from about 30kHz to about 15kHz. Here are some measurements taken with a Clipper Board:

I7000 Value	Output Resolution Signed	Voltage Output Change Per 1bit increment In output register	PWM Frequency	Approximate Ripple Magnitude	Approximate Ripple Frequency
1001	@11 bit	9.9mV	29.4177 KHz	300mV	30 KHz

2002	@12 bit	4.99mV	14.72 KHz	800mV	15 KHz
4004	@13bit	2.49mV	7.36 KHz	2V	7 Khz

How does the ripple affect servo performance? It really depends on the system. For most servo systems the mechanics can't respond anywhere near these frequencies. Some systems with linear amplifiers will affect the performance especially as you lower the PWM frequency and effectively the ripple frequency, i.e. galvanometers, etc. In the overall majority of the servo world, these ripple frequencies will not show in the system due to mechanical and electrical time constants of most systems. This will happen regardless of the amplifier used.

So why is the recommended setup for 30 KHz? The first reason is aesthetics. Nobody wants to put a scope on an output signal and see 1 or 2V of hash. If you increase that frequency, the hash is minimized. The second reason is response of the output with respect to the servo filter. If you increase the output resolution and thus lower the PWM frequency far enough, you will notice some lag in the system from the delays between the output register values actually being picked up by the slower PWM frequency.

For high response systems we suggest using ACC-8ES and a true 18bit DAC. However the filtered PWM technique will be more than adequate for most applications.

How changing I7000 affects other settings in PMAC

I7000 is does not only set the PWM frequency for the PWM outputs, but it also sets the Max Phase Frequency.

*MaxPhase Frequency = 117,964.8 KHz / [2*I7000+3]*

*PWM Frequency = 117,964.8 KHz / [4*I7000+6]*

The Max Phase Frequency is then divided by I7001 to generate the frequency for the phase interrupt and its routines. **If you change I7000, you have to change I7001 to keep the same phase interrupt.**

PHASE Clock Frequency = MaxPhase Frequency / (I7001+1)

The Phase Clock Frequency setting also affects the servo interrupt frequency. **If you change the phase interrupt frequency then you must change I7002 to keep the same servo interrupt.**

Servo Clock Frequency = PHASE Clock Frequency / (I7002+1)

When you change the servo interrupt, you must always change the servo interrupt time – I10 – to match, or all of your timing will be off in PMAC.

*I10 = 838860 8 / (Servo Frequency (KHz)) = 8388608 * ServoTime(msec)*

If you decide to change I7000, be sure to reset Ixx69 to the proper safety setting per the following formula:

*Ixx69 = MaxVolts / 10 * I7000*

Examples:

Default Example:

```
I7000 = 1001
I7001 = 5
I7002 = 3
Ixx69 = 1024
I10   = 3421867
```

*MaxPhase Frequency = 117,964.8 kHz / [2*1001+3] = 58.835 KHz*

*PWM Frequency = 117,964.8 kHz / [4*1001+6] = 29.418 KHz*

PHASE Clock Frequency = MaxPhase Frequency / (5+1) = 9.805 KHz

Servo Clock Frequency = PHASE Clock Frequency / (3+1) = 2.451 KHz

I10 = 8388608 / (2.451471) = 3421867

*Ixx69 = 10V / 10 * I900 = 1001 add headroom to 1024*

To double the resolution, observe the following:

I7000=2002

*MaxPhase Frequency = 117,964.8 KHz / [2*2002+3] = 29.44 KHz*

*PWM Frequency = 117,964.8 KHz / [4*2002+6] = 14.72 KHz*

In order to save headroom on firmware routines that trigger off the phase and servo interrupts, it is best to keep those frequencies about the same as above. Some systems may want higher phase and servo interrupt frequencies for better servo performance, but these default frequencies are typically more than fast enough for many applications. Tuning parameter are discussed elsewhere in this document.

I7001= 29.44 KHz / 19.61KHz - 1 = @0.5 set it at 1 or 14.72 KHz

This is not exactly the same since I7001 is an integer value but the result is close enough for most users. Since we are doing any commutation with a +/-10V signal, it doesn't make that much of a difference. The Servo Frequency we will be able to get close though:

I7002 = 14.72KHz / 4.9 – 1 = 2.004 or 2 which is @ 4.9 KHz

For a 10V max signal output:

Ixx69 = I900 + headroom = 2024

We must set I10 whenever we change the servo clock but since we kept it basically the same, I10 stays pretty much the same. Without rounding it works out to the following:

```
I10 = 8388608 / 4.906613 = 1709653
```

For precise timing within your motion application, it is important not to round off when calculating I10.

Effects of Output Resolution and Servo Frequency on Servo Gains

When changing output resolution and/or servo interrupt timing, tuning parameters will no longer respond the same. The system will have to be tuned again in order to achieve the desired performance. There is an approximate relation of output resolution to servo loop gains. If one were switching an application from a PMAC style 16bit DAC to a Clipper board with default resolution of about 11bits, a change of gains can be expected in order to get similar response. Increasing the servo frequency will require lower proportional gain in order to achieve similar performance.

The max output value of the output command with a 16bit DAC is 32767. With the Clipper set at default parameters, the max output value is 1001. For equal servo interrupt frequencies, the proportional gain on the Clipper would have a proportional gain 1001/32767 or about 1/32 smaller. This is more a rule of thumb than an exact formula. It is always recommended to go through a full tuning procedure when changing output resolution.

If one desires to change servo interrupt frequency in order to have foreground PLCs execute more often then also adjust Ixx60 to keep the gains the same. See the Turbo PMAC Software Reference Manual for a further description of this parameter.

Pulse and Direction (Stepper) Setup

The Clipper must be properly configured to perform PFM stepper motor control functions by setting the I-variables in the following section. Refer to the Turbo PMAC Users and the Turbo Software Reference manuals for complete details on PFM setup and these I-variable assignments.

Multi-Channel Servo IC I-Variables

These I-variables configure global PFM clock settings for all four channels of the addressed servo gate.

I7m03

PFM Clock Frequency: At default this value will be 2258, which is a PFM clock of approximately 10 MHz, (which gives a range about 10 times greater than usually needed). Therefore, this value is not normally changed. Refer to the Turbo Software Reference manual for changing these variables.

I7m04

PFM Pulse Width: The pulse width is specified in PFM clock cycles and has a range of 1 to 255 cycles. The default value is 15. Since the default value of PFM clock is actually set to 9.8304 MHz, the default output pulse width will be 15/9830400 = 1.5258 µS. Note that when the PFM clock values are changed, the PFM pulse width values must be evaluated for proper stepper drive operation.

The user of a typical stepper drive should not need to modify these control variables. However, PFM pulse width should be increased if the stepper drive's input cannot handle the speed of the pulse output. This often occurs with slow opto-couplers used on stepper drive inputs.

Channel-Specific Servo IC I-Variables

There are several hardware setup I-variables for the 12 axes possible with the Clipper controller. These are arranged with IC number (m) and channel number (n). For example, axis 12 would be the fourth channel (n=4) on the third IC (m=2; second ACC-1P). Refer to the Turbo Software Reference Manual for a complete description.

I7mn0 Encoder/Timer n Decode Control

Range: 0 - 15 (Bits 0-3 of channel control word)
Units: None
Default: 7

Caution:

If I7mn0 and I7mn8 are not matched properly, motor runaway will occur.

I7mn0 controls how the input signal for encoder is decoded into counts. This defines how a Turbo PMAC2 interprets the sign and magnitude of a "count". The following are typical settings used with PFM setup:

Value	Feedback type
1	x1 quadrature decode CW
2	x2 quadrature decode CW
3	x4 quadrature decode CW
5	x1 quadrature decode CCW
6	x2 quadrature decode CCW
7	x4 quadrature decode CCW
8	Internal pulse and direction

I7mn0 should be set to 8 to send the output pulses back into the encoder counter – this is the typical electronic loop stepper setup. If an external encoder is used I7mn0 should be set to the appropriate quadrature decode (to match the direction sense of the PFM output). In this case the motor and encoder will behave like a servo-motor and should be tuned accordingly.

I7mn6 Output Mode Select

Range: 0 - 3
Units: None
Default: 0

This I-variable establishes the format of outputs that will be used for each axis. The Clipper has three channels that may be used to control different motor types. The Clipper when in PFM mode uses only the 'C' channel output of any particular axis. A value of 2 or 3 must be used to enable PFM output on channel 'C'. The default value of 3 will set the Clipper to output PFM.

I7mn7 Output n Invert Control

Range: 0 - 3
Units: None
Default: 0

This I-variable establishes a polarity for the output pulses. Changing this variable to a 2 or 3 will cause the output pulses to be active low. The default of 0 for this variable should be sufficient for a typical stepper motor drive interfaces.

I7mn8 Output n PFM Direction Signal Invert Control

Range: 0 - 1
Units: None
Default: 0

Caution:

If I7mn0 and I7mn8 are not matched properly, motor runaway will occur.

The polarity of the direction output is controlled by this I-variable. This output establishes an active low or high output. The default of 0 for this variable should be sufficient for a typical stepper motor drive interfaces.

Encoder Conversion Table I-Variables (I8000 – I8191)

Entries of the encoder conversion table (ECT), using I-variables I8000 – I8191, need to be modified according to the feedback type used: internal pulse and direction for electronic loop stepper control (I7mn0=8) or true encoder feedback for servo loop stepper control (I7mn0=1, 2, 3, 5, 6, 7).

Servo IC #	Chan. 1	Chan. 2	Chan. 3	Chan. 4	Notes
1	I8000=$m78000	I8001=$m78008	I8002=$m78010	I8003=$m78018	Clipper Board
2	I8004=$m78100	I8005=$m78108	I8006=$m78110	I8007=$m78118	First ACC-1P
3	I8008=$m79200	I8009=$m79208	I8010=$m79210	I8011=$m79218	Second ACC-1P
The first hexadecimal digit in the entry: C for no extension of an incremental encoder if using open loop stepper setup (I7mn0 = 8). 0 for 1/T extension of digital incremental encoders if using closed loop stepper setup (I7mn0=1, 2, 3, 5, 6, 7).					

For example, to change the default ECT entry for motor #1 from 1/T extension to no extension:

```
I8000 = $C78000
```

For best results sub-count interpolation (1/T extension) should be shut off. Also the Encoder Conversion Table editor in the PMAC Executive program can also be used to modify the settings.

Be sure to use the **SAVE** command to store the above selections into flash memory.

Motor Setup I-Variables

Several motor I-variables must be set up properly to use PFM signals properly. Most of these are address registers. Typically, motor 1 will use the circuits for axis interface 1, but this is not absolutely necessary. Refer to the Turbo Software Reference Manual for more complete details on I-variable assignments.

Ixx00 Motor xx Activate

This must be set to 1 to activate the PID for motor xx.

Ixx01 Motor xx Commutate Enable

This must be set to 0 for all PFM axes.

Ixx02 Motor xx Command Output Address

This I-variable tells Clipper where to write output data. The following list of addresses is the locations of the C output PFM registers for each channel.

Ixx02 for Clipper base board	Ixx02 for 1st Acc-1P	Ixx02 for 2nd Acc-1P
$078004	$078104	$078204
$07800C	$07810C	$07820C
$078014	$078114	$078214
$07801C	$07811C	$07821C

These values are different from the default values. They must be set properly to operate in PFM mode.

Ixx03 Motor x Position Loop Feedback Address

This I-variable tells PMAC2 where to look to get its position loop feedback. The following list of addresses is the locations of the processed encoder counters for Turbo PMAC2:

I103 = $003501	I903 = $003509
I203 = $003502	I1003 = $00350A
I303 = $003503	I1103 = $00350B
I403 = $003504	I1203 = $00350C
I503 = $003505	I1303 = $00350D
I603 = $003506	I1403 = $00350E
I703 = $003507	I1503 = $00350F
I803 = $003508	I1603 = $003510

These addresses are pointers to the encoder conversion table entries. They point to a set of registers that determine whether sub-count interpolation is used.

Ixx04 Velocity Loop Feedback

This I-variable works very similar to Ixx03. Unless special conditions are desired, Ixx04 is set to the same value as Ixx03 (this is the default).

Ixx08, Ixx09 Motor xx Position and Velocity Loop Scale Factors

These two I-variables present a method of scaling motor activity. They both default to 96 and need not be changed for most applications.

Note:

Changing Ixx08 will have an affect on gain parameter Ixx30.

Ixx11 - Ixx19 Motor x Safety Variables

Refer to the Turbo Software Reference Manual for the settings involved with these variables. The default settings of these variables usually work although they may restrict the motor's speed and/or acceleration.

Ixx20 - Ixx23 Motor Movement Variables

Refer to the Turbo Software Reference Manual for the settings involved with these variables. These variables establish homing and jogging speeds. For initial setup of the Clipper, the default values usually work.

Ixx24 Motor xx Flag Mode Control

Refer to the Turbo Software Reference Manual for the settings involved with this variable. This variable may be set to distinguish how the flags are used for the system. It should be noted that flags might be disabled by prefixing a 2 to the values of the Limit/Home flag addresses. This makes it easier to jog a system during the setup process if motors are uncoupled from their load.

Normally, the default values are acceptable for this I-variable.

Ixx25 Motor xx Flag Address

Refer to the Turbo Software Reference Manual for the settings involved with this variable. Ixx25 tells Turbo PMAC what registers it will access for the overtravel limit inputs, home flag input, amplifier-fault input, and amplifier-enable output for Motor xx. Typically, these are the flags associated with encoder input.

Normally, the default values are acceptable for this I-variable.

Ixx29 Motor xx Output DAC Bias/First Phase Offset

This bias output is used to compensate for input offsets in analog drivers. Set this value to 0 (default) for the PFM mode.

Ixx69 Motor xx Output Command Limit

Range: 0 - 32,767
Units: PFM Register Bits
Default: 20,480

This parameter defines the magnitude of the largest number that may be placed into the PFM register for output pulse frequency. If the Clipper calculates a larger number than Ixx69's value, the number will be clipped to the limiting value.

The result of a limited output value will be a larger following error.

The default value appears to exceed most requirements for stepper motor applications. However, the user may wish to limit the maximum pulse rate near the end of the setup process.

The formula for setting Ixx69 is:

$$Ixx69 = \frac{MaxFreq\ (kHz, MHz)}{PFMCLK\ (kHz, MHz)} X\ 65{,}536$$

The default values for I7m03, I7m04, and Ixx69 will yield an output pulse frequency limit of approximately 3,071,875 PPS (Pulses Per Second). Putting a cap on the pulse output rate may keep the stepper system from exceeding its maximum step rate, which will help keep the motor from losing sync with the Clipper.

Ixx70 Motor xx Number of Commutation Cycles

Stepper motor operation requires that this value be set to 0. The default value is 1, therefore the user must set this value.

Accidentally leaving this I-variable at a non-zero value may result in a motor that dithers when it should be stopped. The reason for this variable to be set to 0 has to do with PMAC2 internal computations.

Motor Servo Gain I-Variables

The Clipper applies its gain formulas the same way it does for a classic servo system. The basic difference with a stepper system is that the typical encoder feedback interface is handled using electronic circuitry rather than a physical encoder.

The Clipper allows the use of both an electronic encoder feedback and/or a physical encoder feedback. When used with an actual physical encoder, the axis should be tuned as if it were a typical servomotor.

The process of tuning the simulated feedback loop is identical to tuning a servomotor with the exception that some of the parameters become more predictable. To create a closed-loop position response with a natural frequency of 25 Hz and a damping ratio of 1 (suitable for almost all systems), use the gain settings as calculated in the following sections.

Ixx30 Motor xx Proportional Gain

To create a closed loop position response with a natural frequency of approximately 25 Hz and a damping ratio of 1, use the following calculation:

$$Ixx30 = \frac{660{,}000}{Ixx08 * PFMCLK(MHz)}$$

Example:
PFMCLK is set to default of 9.83 MHz, and Ixx08 is set to default of 96.

```
Ixx30 = 660,000 / (96 * 9.83) = 700.
```

Ixx31 Motor x Derivative Gain

Derivative Gain is set to 0 because the motor system behaves like a velocity-loop servo drive. This parameter sets the system damping which should be unnecessary.

Ixx32 Motor xx Velocity Feedforward Gain

Use the following equation to establish a value for Ixx32:

```
Ixx32 = 6660 * ServoFreq (kHz)
```

where ServoFreq (kHz) is the frequency of the servo interrupt as established by I7m00, I7m01, and I7m02.

Example:

ServoFreq is set to default of 2.26 kHz (I7m00 = 6527, I7m01 = 0, I7m02 = 3).

```
Ixx32 = 6660 * 2.26 = 15,050.
```

Note:

If Ixx30 were set differently from the above calculation, then Ixx32 would change inversely. For instance, if Ixx30 were twice the above calculation, then Ixx32 would be half its calculation.

Ixx33 Motor xx Integral Gain

Typically, This I-variable should be set to 0. The digital electronic loop does not present offsets or disturbances that need correction in the Clipper.

Ixx33 may be set to force zero steady-state errors, should they be present with electronic encoder feedback.

Ixx34 Motor xx Integration Mode

The default value of 1 is sufficient for this, since usually Ixx33 is set to zero. When Ixx33 is set to 0, this I-variable has no effect.

Ixx35 Motor xx Acceleration Feedforward Gain

Start with this I-variable set to 0. Typically, this value does not need to be changed. However, Ixx35 might be adjusted to compensate for the small time delays created by the electronics when accelerating the stepper.

The effect of adjusting Ixx35 will be to reduce a slight following error during motor acceleration.

Ixx36 - Ixx39 Motor xx Notch Filter Coefficients

These values should be set to their default value of 0. Since filter parameters adjust the way the gains operate due to physical resonance of a system, there is no need to set these I-variables.

In the following examples assume all other settings at factory default ($$$***).

Clipper base board Example

4-axis PFM from base address $078000 motors 1-4. No axis flags are used – amplifier fault and overtravel limits must therefore be disabled.

```
I7016 = 3          ; A and B outputs will be
I7026 = 3          ; DAC and C output will be PFM
I7036 = 3
I7046 = 3

I7010 = 8          ; Internal PFM feedback
I7020 = 8
I7030 = 8
I7040 = 8

I100 = 1           ; Activate motors
I200 = 1
I300 = 1
I400 = 1

I103 = $3501       ; Pos/Vel loop pointers
I104 = $3501
I203 = $3502
I204 = $3502
I303 = $3503
I304 = $3503
```

```
I403 = $3504
I404 = $3504

I102 = $078004     ; Command output to C channel
I202 = $07800C
I302 = $078014
I402 = $07801C

I124 = $120001     ; disable AmpFault and
I224 = $120001     ; MLIM/PLIM (bits 17 and 20 = 1)
I324 = $120001
I424 = $120001

I130 = 700         ; motor #1 gains
I132 = 15050
I133 = 5000

I230 = 700         ; motor #2 gains
I232 = 15050
I233 = 5000

I330 = 700         ; motor #3 gains
I332 = 15050
I333 = 5000

I430 = 700         ; motor #4 gains
I432 = 15050
I433 = 5000

I8000 = $C78000    ; ECT no 1/T extension motors 1-4
I8001 = $C78008
I8002 = $C78010
I8003 = $C78018
I8004 = 0
```

First ACC-1p Example

4-axis PFM from base address $078100 motors 5-8. No axis flags are used – amplifier fault and overtravel limits must therefore be disabled.

```
I7116=3            ; A and B outputs will be
I7126=3            ; DAC and C output will be PFM
I7136=3
I7146=3

I7110 = 8          ; Internal PFM feedback
I7120 = 8
I7130 = 8
I7140 = 8

I500 = 1           ; Activate motors
I600 = 1
I700 = 1
I800 = 1

I503 = $3505       ; Pos/Vel loop pointers
I504 = $3505
I603 = $3506
```

```
I604 = $3506
I703 = $3507
I704 = $3507
I803 = $3508
I804 = $3508

I502 = $078104    ; Command output to C channel
I602 = $07810C
I702 = $078114
I802 = $07811C

I524 = $120001    ; disable AmpFault and
I624 = $120001    ; MLIM/PLIM (bits 17 and 20 = 1)
I724 = $120001
I824 = $120001

I530 = 700        ; motor #5 gains
I532 = 15050
I533 = 5000

I630 = 700        ; motor #6 gains
I632 = 15050
I633 = 5000

I730 = 700        ; motor #7 gains
I732 = 15050
I733 = 5000

I830 = 700        ; motor #8 gains
I832 = 15050
I833=5000

I8004 = $C78100   ; ECT no 1/T extension motors 5-8
I8005 = $C78108
I8006 = $C78110
I8007 = $C78118
I8008 = 0
```

Using Flag I/O as General-Purpose I/O

Either the user flags or other not assigned axes flag on the base board can be used as general-purpose I/O for up to 20 inputs and 4 outputs at 5-24Vdc levels. The indicated suggested M-variables definitions, which are defined in the Software reference, allows accessing each particular line according to the following table:

Flag	Type	Channel Number			
		#1	#2	#3	#4
HOME	5-24 VDC Input	M120	M220	M320	M420
PLIM	5-24 VDC Input	M121	M221	M321	M421
MLIM	5-24 VDC Input	M122	M222	M322	M422
USER	5-24 VDC Input	M115	M215	M315	M415
AENA	5-24 VDC Output	M114	M214	M314	M414

Note:

When using these lines as regular I/O points the appropriate setting of the Ixx24 variable must be used to enable or disable the safety flags feature.

Analog Inputs Setup

The optional analog-to-digital converter inputs are ordered either through Option-12 or Option-2 on the axes expansion board. Each option provides two 12-bit analog inputs with a ±10Vdc range. The M-variables associated with these inputs provided a range of values between +2048 and –2048 for the respective ±10Vdc input range. The following is the software procedure to setup and read these ports.

CPU Analog Inputs

```
I7003 = 1746                ;Set ADC clock frequency at 4.9152 MHz
I7006 = $1FFFFF             ;Clock strobe set for bipolar inputs
M105->Y:$78005,12,12,S      ;ADCIN_1 on JMACH1 connector pin 45
M205->Y:$7800D,12,12,S      ;ADCIN_2 on JMACH1 connector pin 46
```

General-Purpose Digital Inputs and Outputs

The lines on the JOPT general-purpose I/O connector will be mapped into PMAC's address space in register Y:$78400.

Typically, these I/O lines are accessed individually with M-variables. Following is a suggested set of M-variable definitions to use these data lines.

```
M0->Y:$78400,0          ; Digital Output M01
M1->Y:$78400,1          ; Digital Output M02
M2->Y:$78400,2          ; Digital Output M03
M3->Y:$78400,3          ; Digital Output M04
M4->Y:$78400,4          ; Digital Output M05
M5->Y:$78400,5          ; Digital Output M06
M6->Y:$78400,6          ; Digital Output M07
M7->Y:$78400,7          ; Digital Output M08
M8->Y:$78400,8          ; Digital Input MI1
M9->Y:$78400,9          ; Digital Input MI2
M10->Y:$78400,10        ; Digital Input MI3
M11->Y:$78400,11        ; Digital Input MI4
M12->Y:$78400,12        ; Digital Input MI5
M13->Y:$78400,13        ; Digital Input MI6
M14->Y:$78400,14        ; Digital Input MI7
```

```
M15->Y:$78400,15        ; Digital Input MI8
M32->X:$78400,0,8       ; Direction Control bits 0-7  (1=output,  0 = input)
M34->X:$78400,8,8       ; Direction Control bits 8-15 (1=output,  0 = input)
M40->X:$78404,0,24      ; Inversion control (0 = 0V, 1 = 5V)
M42->Y:$78404,0,24      ; J9 port data type control (1 = I/O)
```

In order to properly setup the digital outputs, an initialization PLC must be written scanning through once on power-up/reset, then disabling itself:

```
OPEN PLC1 CLEAR
M32=$FF                 ;BITS 0-8 are assigned as output
M34=$0                  ;BITS 9-16 are assigned as input
M40=$FF00               ;Define inputs and outputs
M42=$FFFF               ;All lines are I/O type
DIS PLC1                ;Disable PLC1 (scanning through once on
                        ;power-up/reset)
CLOSE
```

Note:

After loading this program, set I5=2 or 3 and ENABLE PLC 1.

Thumbwheel Port Digital Inputs and Outputs

The inputs and outputs on the thumbwheel multiplexer port J8 may be used as discrete, non-multiplexed I/O. In this case, these I/O lines can be accessed through M-variables:

```
M40->Y:$78402,8,1             ; SEL0 Output
M41->Y:$78402,9,1             ; SEL1 Output
M42->Y:$78402,10,1            ; SEL2 Output
M43->Y:$78402,11,1            ; SEL3 Output
M44->Y:$78402,12,1            ; SEL4 Output
M45->Y:$78402,13,1            ; SEL5 Output
M46->Y:$78402,14,1            ; SEL6 Output
M47->Y:$78402,15,1            ; SEL7 Output
M48->Y:$78402,8,8,U           ; SEL0-7 Outputs treated as a byte
M50->Y:$78402,0,1             ; DAT0 Input
M51->Y:$78402,1,1             ; DAT1 Input
M52->Y:$78402,2,1             ; DAT2 Input
M53->Y:$78402,3,1             ; DAT3 Input
M54->Y:$78402,4,1             ; DAT4 Input
M55->Y:$78402,5,1             ; DAT5 Input
M56->Y:$78402,6,1             ; DAT6 Input
M57->Y:$78402,7,1             ; DAT7 Input
M58->Y:$78402,0,8,U           ; DAT0-7 Inputs treated as a byte
```

Setup of a Fifth Motor Using Opt-12 on the Clipper Board

The DSPGATE2A supplemental channels are set with I6800-6807. Set these to the same values as specified for the filtered PWM outputs (leave I6804-I6807 at default):

```
I6800  = 1001                 ; PWM frequency 29.4kHz, PWM 1-4
I6801  = 5                    ; Phase Clock 9.8kHz
I6802  = 3                    ; Servo frequency 2.45kHz
I6803  = 1746                 ; ADC frequency
I68n6  = 0                    ; Output mode: PWM
Ixx69  = 1001                 ; DAC limit 10Vdc
```

```
;n =  supplementary channels 1 and 2
;xx = motor number from 5 to 32
```

The encoder decode I-variables are I68n0-68n9 (n=1 or 2). Set these for your encoders as normal. Note there are no direct inputs for flags so capture I-variables are not used. The Output Command Registers (Ixx02) now must point to the DSPGATE2A 3rd Channel Outputs at $78414 and $7841C first and second supplemental registers respectively. The addresses of the DSPGATE2A Counters/Timers used in the encoder conversion table are $78410 and $78418 first and second supplementary registers respectively. When using the OPT-12 filtered PWM DAC on the hand-wheel port use the second output at $7841C. The encoder counter registers are at:

```
Mxx->X:$78411,0,24,s // first counter register
Mxx->X:$78419,0,24,s // second counter register
```

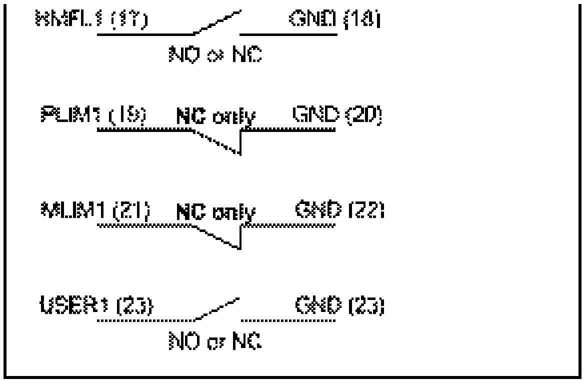
Flags access through JOPTO port.

The OPT-12 channel may also access its machine I/O such as the overtravels, home, and fault flags for one motor only. This is done through the JOPTO I/O lines (J9), by changing their function so that they would act as Home / Pos / Neg / User flags for only the first channel of the two supplemental channels; there are no flags for the second channel since these pins of the general I/O port of the PMAC2 gate2 are not brought out in this board.

The following example sets up the OPT-12 for two motors (#9 and #10) of pulse and direction control with full machine I/O on motor #9.

Command Output	Port	Pin
PUL1+	J10	11
PUL1-	J10	12
DIR1+	J10	13
DIR1-	J10	14

Flag Inputs	Port	Pin
HMFL1	J9	17
PLIM1	J9	19
MLIM1	J9	21
USER1	J9	23
Wiring Example:		



Jumpers	Setting
E16	Install
E17	Remove

Settings:

```
M32->X:$78400,0,8  ; Direction Control bits 0-7 (1=output, 0 = input)
M34->X:$78400,8,8  ; Direction Control bits 8-15 (1=output, 0 = input)
M40->X:$78404,0,24 ; Inversion control (0 = 0V, 1 = 5V)
M42->Y:$78404,0,24 ; J9 port data type control (1 = I/O)
```

Power-up PLC:

```
Open plc 1 clear
M32=$00             ;IO 1/8 inputs
M34=$FF             ;IO 9/16 Outputs
M40=$0              ; Do not invert anything
M42=$FF0F           ; 1~4 GP I/O   5~8 flags   9~16 GP I/O
Disable plc 1
Close


I925=$78410         ;mot #9
I1025=$78418        ;mot #10 for this motor use I1024=$520001 to disable amp
                    ;fault and overtravels
I902 = $78414       ;mot #9
I1002 = $7841C      ;mot #10
I6810 = 8           ; Internal pulse and direction     ;mot #9
I6816= 3            ; PFM on C
I6820 = 8           ; Internal pulse and direction     ;mot #10
I6826= 3            ; PFM on C
```

For motor #9 use I903 and I904 appropriate ECT entry performing a “No extension of quadrature encoder” pointing to $78410. For motor #10 use I1003 and I1004 appropriate ECT entry performing a “No extension of quadrature encoder” pointing to $78418.

CLIPPER OPTION-11A APPLICATION NOTE

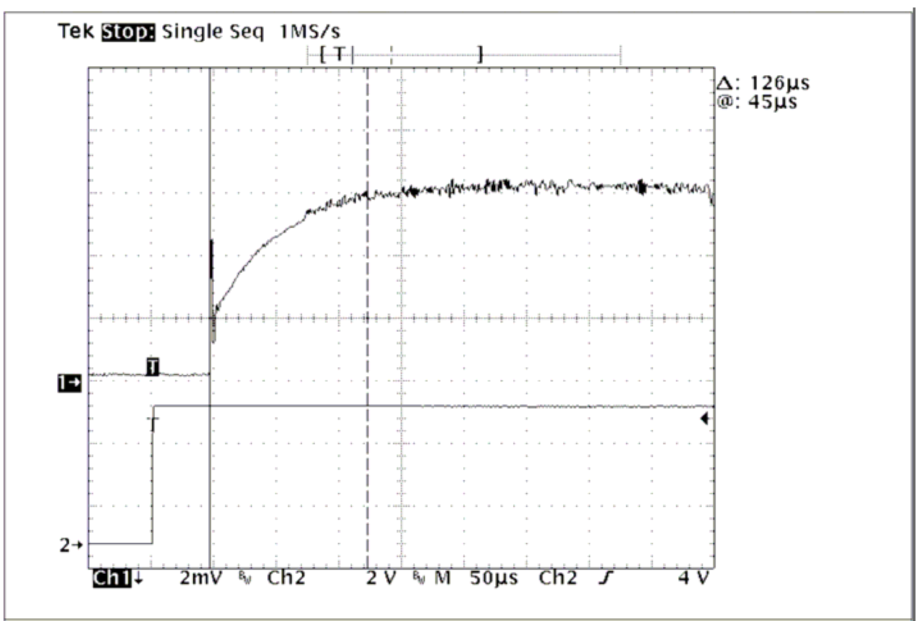
Clipper's Option 11 consists of a programmable lattice chip which can be programmed based upon customer's requirements. The main objective for this option is to be used as a laser controller. Different programs can be loaded in this chip based upon customers requirements and each code will be designated an alpha-numeric suffix after options number if the code is developed by Delta Tau and can be ordered at a later time with the same suffix.

Clipper's option-11A is developed as a general command signal needed for CO2 lasers. Usually CO2 lasers require a few digital I/O signals in order to control the status and mode of the laser and a control signal, which based upon the signal features controls the output power of the laser. The laser can be in a few different modes:

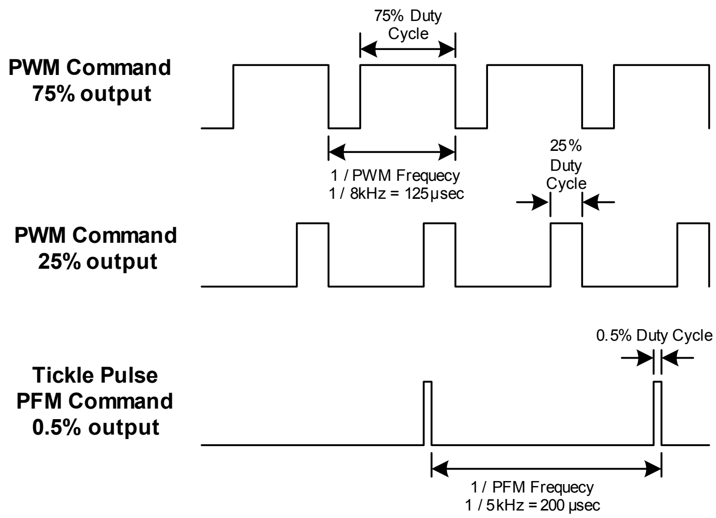
- Disabled
- Standby
- Active

Controlling between disabled mode and other modes is usually done through a digital output, either directly if the device is TTL level or it would have to be done through a relay system. The difference between the "Standby" mode and "Active" mode is because of the signal type and shape. Usually in order to control the output power of the laser, a PWM (Pulse Width Modulation) signal is used and the positive duty cycle of the signal indicates the output power of the laser, varying from 0 to 100%. However, in order to ensure immediate response from the laser when an output is required, the gas needs to be kept ionized. This can be achieved by outputting a PFM signal. The frequency and duty cycle required for each of these modes differs based on the laser model and size and should be adjusted accordingly.

Assuming the following graph is the laser output response to a step command, the frequency of the modulation can be selected.



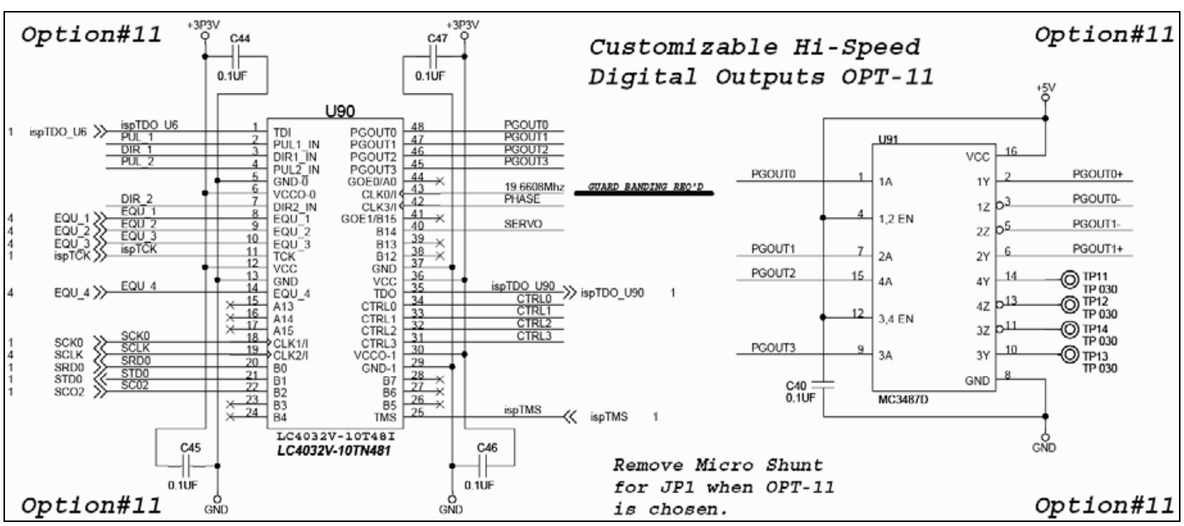
As you can see, the rise time for the laser is about 126µsec. This means in order for our modulation to fully cover the 0 to 100% range of the output, the frequency should be set close to 1/126µsec = 7936 Hz or 8kHz. The tickle pulse is required in order to reduce the time between the change of command to PWM and actual output of the laser. However calculating the signal requirements for the Tickle pulse is dependent on the laser and differs for different manufacturers. For example the laser shown in the above graph, requires a 5kHz signal with 0.5% duty cycle as its Tickle pulse.



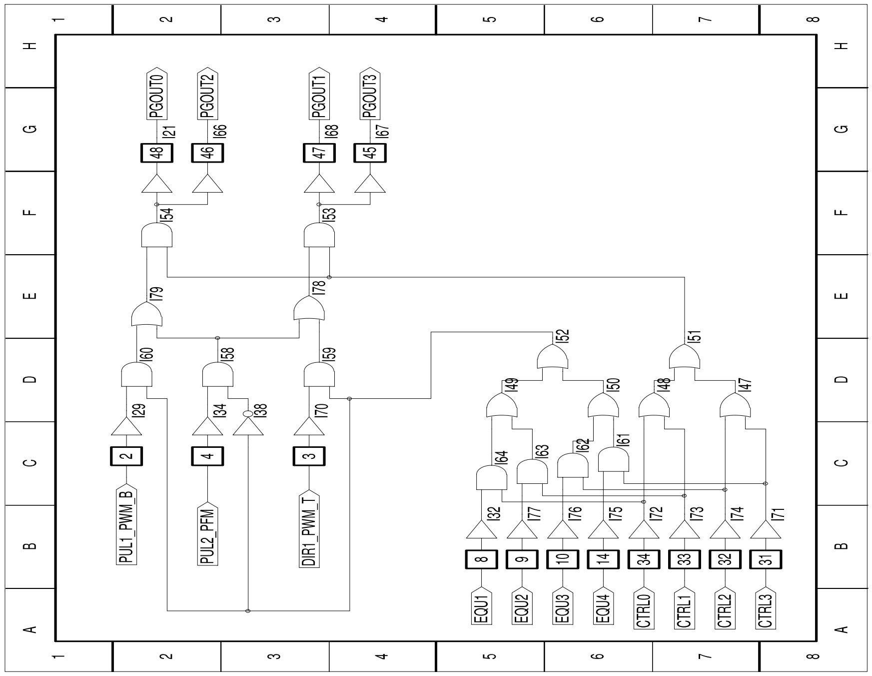
In the next section we will use this laser specific information to set up Option-11A. Please note that the values and settings here are just an example and your values might be different. Please refer to your laser documentation or contact the manufacturer for detailed information about your specific laser.

Understanding Option-11A Capabilities

Option-11A has been programmed to include a few logical gates controlling the output signals. In general there are a few signals available from the Clipper as inputs to the Lattice chip.



The following logic circuit is programmed as the Option-11A into the Lattice chip:



As you can see, the idea is to switch the output between PWM_B signal and PFM signal based upon either of the EQU outputs. EQU outputs are fast responding outputs which can either be activated manually or based upon position compare feature of the PMAC. CTRL outputs control which of the EQUs or what combination of EQUs will be used to control the output mode.

As an example, if a user wants to use EQU1 to switch between the Tickle pulse (PFM) and Output mode (PWM), then CTRL0 must be turned on. As a safety measure, no output will be generated unless at least one of the CTRL outputs is set to 1.

Clock Settings

The clock used for PWM is related to the Max Phase Clock, which is used to generate the Phase and Servo clock, selecting an appropriate PWM frequency which satisfies both the laser requirements and servo requirements can be challenging, especially if user is trying to use Filtered PWM outputs as DAC outputs for amplifier commands.

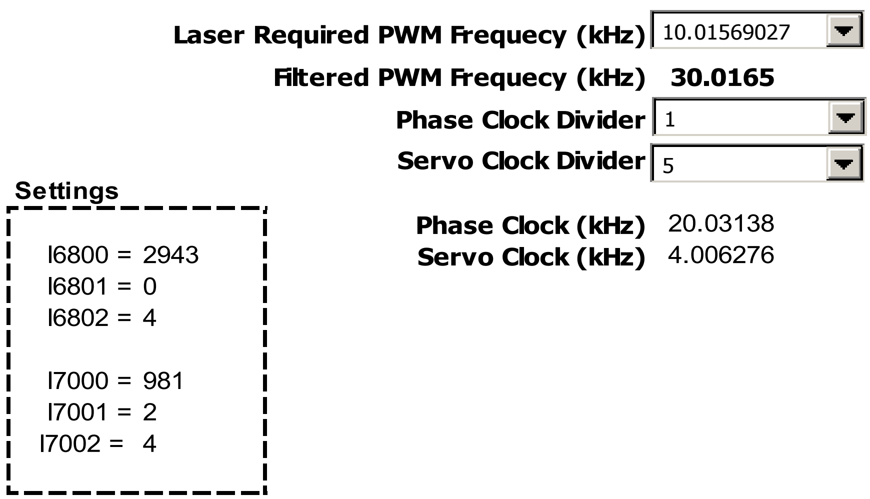
In most of the applications, the PWM frequency is a set value and is not change at all, however in some applications this is not the case and the PWM has to change based upon the material being cut or worked on. If you are required to change the PWM based upon the material, you won’t be able to use the filtered PWM outputs directly available from the Clipper board and you have to use ACC-8ES (Analog Servo Interface) in order to get the DAC outputs for servo amplifier commands.

If the PWM frequency required for the laser is less than 30kHz required for the Filtered PWM outputs, which mostly is the case, then the PWM frequency of the DSPGate1 (30kHz) should be divisible by PWM frequency of the DSPGate2 which is used for generating the laser output signal. The clock source on the Clipper, by default, is the DSPGate1, which in this case needs to be changed to DSPGate2. In order to change the clock source, the following steps should be followed carefully in exact explained order or else you may cause a watchdog.

Switching the clock source:

1. Set **I19=6807**.
2. Issue **SAVE** and **$$$**.
3. Set **I6807=0** and **I7007=3** on the same line.
4. Issue **SAVE** and **$$$**.

This will change the clock source from DSPGate1 to DSPGate2. Once the clock source is switched, the following settings will give you different PWM frequencies on the laser output while keeping the 30 kHz PWM requirement for Filtered PWM outputs:



Controlling the output

This section includes the memory address settings that you would need in order to change the PWM duty cycle, PFM frequency , EQU output mode and EQU selection.

Note:

Please Set **I28=1** and issue a **SAVE** and **$$$** before the following settings become active. The I28=1 will disable the DISPLAY output port which in this case will over-write the CTRL outputs.

```
#define CTRL0   M7000
#define CTRL1   M7001
#define CTRL2   M7002
#define CTRL3   M7003

#define PWM     M7004
#define PFM     M7005

#define CTRL_TYP  M7006
#define CTRL_INV  M7007
#define CTRL_DAT  M7008
#define CTRL_DIR  M7009

#define PWM_CMD_VAL  M7010
#define PFM_CMD_VAL  M7011

#define EQU1_ON   M112=1M111=1
#define EQU1_OFF  M112=0M111=1
#define EQU2_ON   M212=1M211=1
#define EQU2_OFF  M212=0M211=1
```

```
#define EQU3_ON   M312=1M311=1
#define EQU3_OFF  M312=0M311=1
#define EQU4_ON   M412=1M411=1
#define EQU4_OFF  M412=0M411=1

CTRL_TYP->Y:$078407,8,4
CTRL_INV->X:$078407,8,4
CTRL_DAT->Y:$078403,8,4
CTRL_DIR->X:$078403,8,4

PWM_CMD_VAL->Y:$078414,8,16,S
PFM_CMD_VAL->Y:$07841C,0,24,S

M111->X:$078005,11 ; ENC1 compare initial state write enable
M112->X:$078005,12 ; ENC1 compare initial state
M116->X:$078000,9 ; ENC1 compare output value

M211->X:$07800D,11 ; ENC2 compare initial state write enable
M212->X:$07800D,12 ; ENC2 compare initial state
M216->X:$078008,9 ; ENC2 compare output value

M311->X:$078015,11 ; ENC3 compare initial state write enable
M312->X:$078015,12 ; ENC3 compare initial state
M316->X:$078010,9 ; ENC3 compare output value

M411->X:$07801D,11 ; ENC4 compare initial state write enable
M412->X:$07801D,12 ; ENC4 compare initial state
M416->X:$078018,9 ; ENC4 compare output value

Open PLC 1 Clear
CTRL_DIR=$F
CTRL_DAT=$1
CTRL_TYP=$F
CTRL_INV=$0
I6816=0   ;PWM OUTPUT ON 1st Supplimental Channel
I6826=3   ;PFM Output on 2nd Supplimental Channel
Disable PLC 1
Close

PWM_CMD_VAL = 0           ; represents 50% duty cycle
PFM_CMD_VAL = 3000        ; Changes PFM frequency
```

Based upon the settings above you can change the PWM duty cycle by changing the value of `PWM_CMD_VAL` and the PFM frequency by changing `PFM_CMD_VAL`. The duty cycle of the PFM signal however will be changed based upon I6804 and I6803 settings.

PFM width = I6804 / PFM Clock

The EQU can be turned on and off manually to switch the output mode:

EQU on: M112=1 M111=0

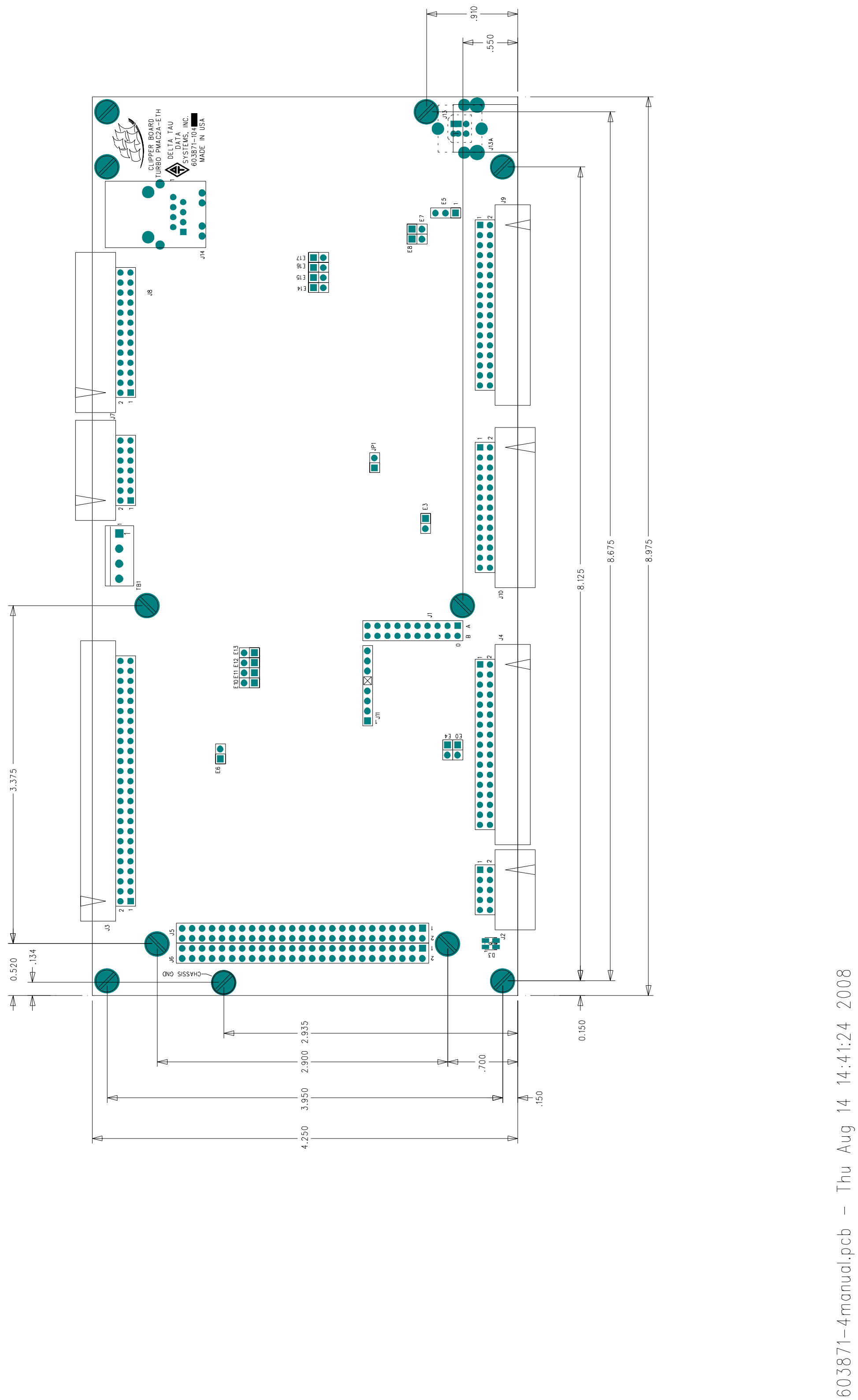
EQU off: M112=0 M111=0

Also, Position Compare function can be used to control the EQU output. Please refer to Turbo Users Manual for detailed information about position compare functionality and settings.

HARDWARE REFERENCE SUMMARY

The following information is based on the Clipper Board, part number 603871.

Board Dimensions and Layout



Layout for REV-103 and REV-104

Connectors and Indicators

J2 - Serial Port (JRS232 Port)

This connector allows communicating with PMAC from a host computer through a RS-232 port. Delta Tau provides the Accessory 3L cable that connects the PMAC to a DB-9 connector.

1. 10-pin female flat cable connector T&B Ansley P/N 609-1041
2. Standard flat cable stranded 10-wire T&B Ansley P/N 171-10

J3 - Machine Connector (JMACH1 Port)

The primary machine interface connector is JMACH1, labeled J3 on the PMAC. It contains the pins for four channels of machine I/O: analog outputs, incremental encoder inputs, amplifier fault and enable signals and power-supply connections.

1. 50-pin female flat cable connector T&B Ansley P/N 609-5041
2. Standard flat cable stranded 50-wire T&B Ansley P/N 171-50
3. Phoenix varioface module type FLKM 50 (male pins) P/N 22 81 08 9

J4 - Machine Connector (JMACH2 Port)

This machine interface connector is labeled JMACH2 or J4 on the PMAC. It contains the pins for four channels of machine I/O: end-of-travel input flags, home flag and pulse-and-direction output signals. In addition, the B_WDO output allows monitoring the state of the Watchdog safety feature.

1. 34-pin female flat cable connector T&B Ansley P/N 609-3441
2. Standard flat cable stranded 34-wire T&B Ansley P/N 171-34
3. Phoenix varioface module type FLKM 34 (male pins) P/N 22 81 06 3

J7 - Machine Connector (JMACH3 Port)

This machine interface connector is labeled JMACH3 or J7 on the PMAC. It contains the pins for four channels of U, V, and W flags normally used for hall device commutation.

1. 14-pin female flat cable connector Delta Tau P/N 014-R00F14-0K0, T&B Ansley P/N 609-1441
2. 171-14 T&B Ansley standard flat cable stranded 14-wire
3. Phoenix varioface modules type FLKM14 (male pins) P/N 22 81 02 1

J8 - Thumbwheel Multiplexer Port (JTHW Port)

The Thumbwheel Multiplexer Port, or Multiplexer Port, on the JTHW connector has eight input lines and eight output lines. The output lines can be used to multiplex large numbers of inputs and outputs on the port, and Delta Tau provides accessory boards and software structures (special M-variable definitions) to capitalize on this feature. Up to 32 of the multiplexed I/O boards may be daisy-chained on the port, in any combination.

1. 26-pin female flat cable connector T&B Ansley P/N 609-2641
2. Standard flat cable stranded 26-wire T&B Ansley P/N 171.26
3. Phoenix varioface module type FLKM 26 (male pins) P/N 22 81 05 0

J9 - General-Purpose Digital Inputs and Outputs (JOPT Port)

Acc-1P's JOPT connector provides eight general-purpose digital inputs and eight general-purpose digital outputs. Each input and each output has its own corresponding ground pin in the opposite row. The 34-pin connector was designed for easy interface to OPTO-22 or equivalent optically isolated I/O modules. Delta Tau's Acc-21F is a six-foot cable for this purpose.

1. 34-pin female flat cable connector T&B Ansley P/N 609-3441
2. Standard flat cable stranded 34-wire T&B Ansley P/N 171-34

3. Phoenix varioface module type FLKM 34 (male pins) P/N 22 81 06 3

J10 – Handwheel and Pulse/Dir Connector (JHW/PD Port)

This connector is labeled JHW/PD or J10 on the PMAC. It provides pins for the two channels of Quadrature encoder inputs and Pulse and direction (PFM or PWM) output pairs from the DSPGate2 supplemental channels 1* and 2*.

1. 26-pin female flat cable connector T&B Ansley P/N 609-2641
2. Standard flat cable stranded 26-wire T&B Ansley P/N 171.26
3. Phoenix varioface module type FLKM 26 (male pins) P/N 22 81 05 0

J12 – Ethernet Communications Port

This connector provides access to the Ethernet communications feature. See the Machine Connections chapter for details on using this port.

J13 – USB Communications Port

This connector provides access to the USB communications feature. See the Machine Connections chapter for details on using this port.

JP11 – OPT-11 Shunt

Not present if OPT-11 is installed. For internal use only.

TB1 – Power Supply Terminal Block (JPWR Connector)

This terminal block is the power supply connector for the board.

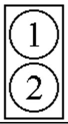
1. 4-pin terminal block, 0.150 pitch

LED Indicators

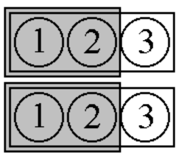
D3: This is a dual color LED. When this LED is green, it indicates that power is applied to the +5V input when this LED is red, it indicates that the watchdog timer has tripped.

E-POINT JUMPER DESCRIPTIONS

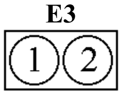
E0: Forced Reset Control

E Point and Physical Layout	Location	Description	Default
E0 		Factory use only; the board will not operate with E0 installed.	No jumper

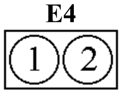
E1 – E2: Serial Port Selection (rev 102 and below only)

E Point and Physical Layout	Location	Description	Default
E1  E2		These jumpers select the target CPU for the serial port as either the main PMAC CPU or the Ethernet CPU (change IP address). Both jumpers must be set the same. • 1-2 for Main CPU • 2-3 for Ethernet CPU	1-2 Jumper installed

E3: Normal/Re-Initializing Power-Up/Reset

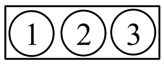
E Point and Physical Layout	Location	Description	Default
E3 		Jump pin 1 to 2 to re-initialize on power-up/reset, loading factory default settings. Remove jumper for normal power-up/reset, loading user-saved settings.	No jumper installed

E4: Watchdog Disable Jumper

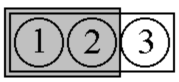
E Point and Physical Layout	Location	Description	Default
E4 		Jump pin 1 to 2 to disable Watchdog timer (for test purposes only). Remove jumper to enable Watchdog timer.	No jumper

E5: Reserved for factory use only

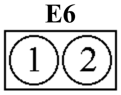
Version 102 and higher

E Point and Physical Layout	Location	Description	Default
E5 		For factory use only; the board will not communicate via Ethernet or USB if jumper E5 is installed.	No Jumper installed

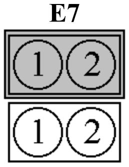
Version 101 and lower

E Point and Physical Layout	Location	Description	Default
E5 		For factory use only; the board will not communicate via Ethernet unless Jumper is installed on pins 1 to 2.	1-2 Jumper installed

E6: ADC Inputs Enable

E Point and Physical Layout	Location	Description	Default
E6 		Jump pin 1 to 2 to enable the Option-12 ADC inputs. Remove jumper to disable the ADC inputs, which might be necessary for reading current feedback signals from digital amplifiers.	No jumper

E7 – E8: Power-Up State Jumpers

E Point and Physical Layout	Location	Description	Default
E7  **E8**		E7 is the reset on power jumper for the USB/EtherNet CPU, remove before power cycle to reset. E8 is the USB/EtherNet CPU write protect jumper. To enable write protect (no IP address change allowed) remove jumper E8.	E7 jumper installed. E8 jumper not installed.

E10 – E12: Power-Up State Jumpers

E Point and Physical Layout	Location	Description	Default
E10 1 2 1 2 1 2 E12		Remove jumper E10; Jump E11; Jump E12; To read flash IC on power-up/reset Other combinations are for factory use only; the board will not operate in any other configuration.	No E10 jumper installed; Jump E11 and E12

E13: Power-Up/Reset Load Source

E Point and Physical Layout	Location	Description	Default
E13 1 2		Jump pin 1 to 2 to reload firmware through serial or bus port. Remove jumper for normal operation.	No jumper

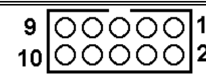
E14- E17: Ports Direction Control

E Point and Physical Layout	Location	Description	Default
E14 1 2		Install jumper to make DATx lines inputs. No jumper to make DATx lines outputs.	Jumper installed
E15 1 2		Install jumper to make SELx lines inputs. No jumper to make SELx lines outputs.	No jumper
E16 1 2		Install jumper to make MOx lines inputs. No jumper to make MOx lines outputs.	No jumper
E17 1 2		Install jumper to make MIx lines inputs. No jumper to make MIx lines outputs.	Jumper installed

CONNECTOR PINOUTS

J2 (JRS232) Serial Port Connector

(10-PIN CONNECTOR)

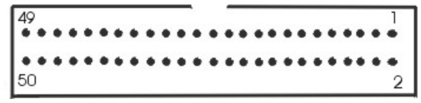


Front View

Pin#	Symbol	Function	Description	Notes
1	PHASE	Output	Phasing Clock	
2	DTR	Bidirect	Data Terminal Ready	Tied to "DSR"
3	TXD/	Output	Send Data	Host receive data
4	CTS	Input	Clear to Send	Host ready bit
5	RXD/	Input	Receive Data	Host transmit data
6	RTS	Output	Request to Send	PMAC ready bit
7	DSR	Bidirect	Data Set Ready	Tied to "DTR"
8	SERVO	Output	Servo Clock	
9	GND	Common	Digital Common	
10	+5V	Output	+5Vdc Supply	Power supply out

J3 (JMACH1): Machine Port Connector

(50-Pin Header)

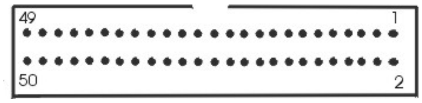


Top View

Pin#	Symbol	Function	Description	Notes
1	+5V	Output	+5V Power	For encoders, 1
2	+5V	Output	+5V Power	For encoders, 1
3	GND	Common	Digital Common	For encoders, 1
4	GND	Common	Digital Common	For encoders, 1
5	CHA1	Input	Encoder A Channel Positive	2
6	CHA2	Input	Encoder A Channel Positive	2
7	CHA1/	Input	Encoder A Channel Negative	2,3
8	CHA2/	Input	Encoder A Channel Negative	2,3
9	CHB1	Input	Encoder B Channel Positive	2
10	CHB2	Input	Encoder B Channel Positive	2
11	CHB1/	Input	Encoder B Channel Negative	2,3
12	CHB2/	Input	Encoder B Channel Negative	2,3
13	CHC1	Input	Encoder C Channel Positive	2
14	CHC2	Input	Encoder C Channel Positive	2
15	CHC1/	Input	Encoder C Channel Negative	2,3
16	CHC2/	Input	Encoder C Channel Negative	2,3
17	CHA3	Input	Encoder A Channel Positive	2
18	CHA4	Input	Encoder A Channel Positive	2
19	CHA3/	Input	Encoder A Channel Negative	2,3
20	CHA4/	Input	Encoder A Channel Negative	2,3
21	CHB3	Input	Encoder B Channel Positive	2
22	CHB4	Input	Encoder B Channel Positive	2
23	CHB3/	Input	Encoder B Channel Negative	2,3
24	CHB4/	Input	Encoder B Channel Negative	2,3
25	CHC3	Input	Encoder C Channel Positive	2
26	CHC4	Input	Encoder C Channel Positive	2
27	CHC3/	Input	Encoder C Channel Negative	2,3
28	CHC4/	Input	Encoder C Channel Negative	2,3
29	DAC1	Output	Analog Output Positive 1	4
30	DAC2	Output	Analog Output Positive 2	4
31	DAC1/	Output	Analog Output Negative 1	4,5
32	DAC2/	Output	Analog Output Negative 2	4,5
33	AENA1/	Output	Amplifier-Enable 1	
34	AENA2/	Output	Amplifier -Enable 2	
35	FAULT1/	Input	Amplifier -Fault 1	6
36	FAULT2/	Input	Amplifier -Fault 2	6
37	DAC3	Output	Analog Output Positive 3	4
38	DAC4	Output	Analog Output Positive 4	4
39	DAC3/	Output	Analog Output Negative 3	4,5

J3 JMACH1 (50-Pin Header)

(Continued)



Top View

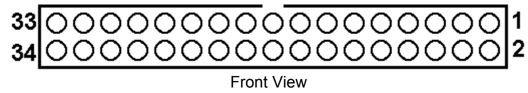
Pin#	Symbol	Function	Description	Notes
40	DAC4/	Output	Analog Output Negative 4	4,5
41	AENA3/	Output	Amplifier -Enable 3	
42	AENA4/	Output	Amplifier -Enable 4	
43	FAULT3/	Input	Amplifier -Fault 3	6
44	FAULT4/	Input	Amplifier -Fault 4	6
45	ADCIN_1	Input	Analog Input 1	Option-12 required
46	ADCIN_2	Input	Analog Input 2	Option-12 required
47	FLT_FLG_V	Input	Amplifier Fault pull-up V+	
48	GND	Common	Digital Common	
49	+12V	Input	DAC Supply Voltage	7
50	-12V	Input	DAC Supply Voltage	7

The J3 connector is used to connect PMAC to the first 4 channels (Channels 1, 2, 3, and 4) of servo amps and encoders.

Note 1: These lines can be used as +5V power supply inputs to power PMAC's digital circuitry.
Note 2: Referenced to digital common (GND). Maximum of ±12V permitted between this signal and its complement.
Note 3: Leave this input floating if not used (i.e. digital single-ended encoders).
Note 4: ±10V, 10 mA max, referenced to common ground (GND).
Note 5: Leave floating if not used. Do not tie to GND.
Note 6: Functional polarity controlled by variable Ixx24. Must be conducting to 0V (usually GND) to produce a 0 in PMAC software. Automatic fault function can be disabled with Ixx24.
Note 7: Can be used to provide input power when the TB1 connector is not being used.

J4 (JMACH2): Machine Port CPU Connector

(34-Pin Header)



Pin#	Symbol	Function	Description	Notes
1	FLG_1_2_V	Input	Flags 1-2 Pull-Up	
2	FLG_3_4_V	Input	Flags 3-4 Pull-Up	
3	GND	Common	Digital Common	
4	GND	Common	Digital Common	
5	HOME1	Input	Home-Flag 1	10
6	HOME2	Input	Home-Flag 2	10
7	PLIM1	Input	Positive End Limit 1	8,9
8	PLIM2	Input	Positive End Limit 2	8,9
9	MLIM1	Input	Negative End Limit 1	8,9
10	MLIM2	Input	Negative End Limit 2	8,9
11	USER1	Input	User Flag 1	
12	USER2	Input	User Flag 2	
13	PUL_1	Output	Pulse Output 1	
14	PUL_2	Output	Pulse Output 2	
15	DIR_1	Output	Direction Output 1	
16	DIR_2	Output	Direction Output 2	
17	EQU1	Output	Encoder Comp-Equal 1	
18	EQU2	Output	Encoder Comp-Equal 2	
19	HOME3	Input	Home-Flag 3	10
20	HOME4	Input	Home-Flag 4	10
21	PLIM3	Input	Positive End Limit 3	8,9
22	PLIM4	Input	Positive End Limit 4	8,9
23	MLIM3	Input	Negative End Limit 3	8,9
24	MLIM4	Input	Negative End Limit 4	8,9
25	USER3	Input	User Flag 3	
26	USER4	Input	User Flag 3	
27	PUL_3	Output	Pulse Output 3	
28	PUL_4	Output	Pulse Output 4	
29	DIR_3	Output	Direction Output 3	
30	DIR_4	Output	Direction Output 4	
31	EQU3	Output	Encoder Comp-Equal 3	
32	EQU4	Output	Encoder Comp-Equal 4	
33	B_WDO	Output	Watchdog Out	Indicator/driver
34	No Connect			

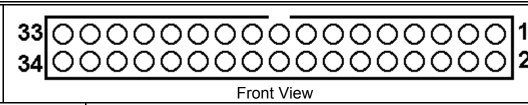
Note 8: Pins marked *PLIMn* should be connected to switches at the *positive* end of travel. Pins marked *MLIMn* should be connected to switches at the *negative* end of travel.

Note 9: Must be conducting to 0V (usually GND) for PMAC to consider itself not into this limit. Automatic limit function can be disabled with Ixx24.

Note 10: Functional polarity for homing or other trigger use of HOMEn controlled by Encoder/Flag Variable I70n2. HMFLn selected for trigger by Encoder/Flag Variable I70n3. Must be conducting to 0V (usually GND) to produce a 0 in PMAC software.

J7 (JMACH3): Machine Port

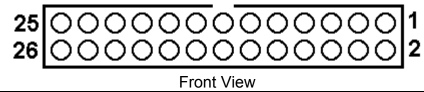
(14-Pin Header)



Pin#	Symbol	Function	Description	Notes
1	GND	Common	Digital Common	
2	GND	Common	Digital Common	
3	CHU1+	Input	U-Flag Channel 1	
4	CHU2+	Input	U-Flag Channel 2	
5	CHV1+	Input	V-Flag Channel 1	
6	CHV2+	Input	V-Flag Channel 2	
7	CHW1+	Input	W-Flag Channel 1	
8	CHW2+	Input	W-Flag Channel 2	
9	CHU3+	Input	U-Flag Channel 3	
10	CHU4+	Input	U-Flag Channel 4	
11	CHV3+	Input	V-Flag Channel 3	
12	CHV4+	Input	V-Flag Channel 4	
13	CHW3+	Input	W-Flag Channel 3	
14	CHW4+	Input	W-Flag Channel 4	

J8 (JTHW): Multiplexer Port Connector

(26-Pin Connector)



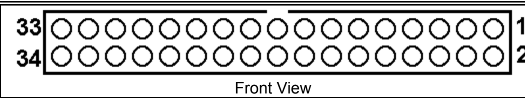
Pin#	Symbol	Function	Description	Notes
1	GND	Common	PMAC Common	
2	GND	Common	PMAC Common	
3	DAT0	Input	Data-0 Input	Data input from multiplexed accessory
4	SEL0	Output	Select-0 Output	Multiplexer select output
5	DAT1	Input	Data -1 Input	Data input from multiplexed accessory
6	SEL1	Output	Select -1 Output	Multiplexer select output
7	DAT2	Input	Data -2 Input	Data input from multiplexed accessory
8	SEL2	Output	Select -2 Output	Multiplexer select output
9	DAT3	Input	Data -3 Input	Data input from multiplexed accessory
10	SEL3	Output	Select -3 Output	Multiplexer select output
11	DAT4	Input	Data -4 Input	Data input from multiplexed accessory
12	SEL4	Output	Select -4 Output	Multiplexer select output
13	DAT5	Input	Data -5 Input	Data input from multiplexed accessory
14	SEL5	Output	Select -5 Output	Multiplexer select output
15	DAT6	Input	Data -6 Input	Data input from multiplexed accessory
16	SEL6	Output	Select -6 Output	Multiplexer select output
17	DAT7	Input	Data -7 Input	Data input from multiplexed accessory
18	SEL7	Output	Select -7 Output	Multiplexer select output
19	N.C.	N.C.	No Connection	
20	GND	Common	PMAC Common	
21	N.C.	N.C.	No Connection	
22	GND	Common	PMAC Common	
23	N.C.	N.C.	No Connection	
24	GND	Common	PMAC Common	
25	+5V	Output	+5VDC Supply	Power supply out
26	INIT-	Input	PMAC Reset	Low is Reset

The JTHW multiplexer port provides 8 inputs and 8 outputs at TTL levels. While these I/O can be used in unmultiplexed form for 16 discrete I/O points, most users will utilize PMAC software and accessories to use this port in multiplexed form to greatly multiply the number of I/O that can be accessed on this port. In multiplexed form, some of the SELn outputs are used to select which of the multiplexed I/O are to be accessed.

The direction of the input and output lines on this connector are set by jumpers E14 and E15. If E14 is removed or E15 is installed then the multiplexing feature if the JTHW port cannot be used.

J9 (JOPT): I/O Port Connector

(34-Pin Connector)

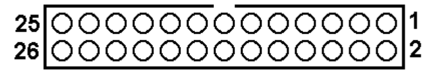


Pin#	Symbol	Function	Description	Notes
1	MI8	Input	Machine Input 8	12, 13
2	GND	Common	PMAC Common	
3	MI7	Input	Machine Input 7	12, 13
4	GND	Common	PMAC Common	
5	MI6	Input	Machine Input 6	12, 13
6	GND	Common	PMAC Common	
7	MI5	Input	Machine Input 5	12, 13
8	GND	Common	PMAC Common	
9	MI4	Input	Machine Input 4	12, 13
10	GND	Common	PMAC Common	
11	MI3	Input	Machine Input 3	12, 13
12	GND	Common	PMAC Common	
13	MI2	Input	Machine Input 2	12, 13
14	GND	Common	PMAC Common	
15	MI1	Input	Machine Input 1	12, 13
16	GND	Common	PMAC Common	
17	MO8	Output	Machine Output 8	11, 13
18	GND	Common	PMAC Common	
19	MO7	Output	Machine Output 7	11, 13
20	GND	Common	PMAC Common	
21	MO6	Output	Machine Output 6	11, 13
22	GND	Common	PMAC Common	
23	MO5	Output	Machine Output 5	11, 13
24	GND	Common	PMAC Common	
25	MO4	Output	Machine Output 4	11, 13
26	GND	Common	PMAC Common	
27	MO3	Output	Machine Output 3	11, 13
28	GND	Common	PMAC Common	
29	MO2	Output	Machine Output 2	11, 13
30	GND	Common	PMAC Common	
31	MO1	Output	Machine Output 1	11, 13
32	GND	Common	PMAC Common	
33	+5	Output	+5 Power I/O	
34	GND	Common	PMAC Common	

This connector provides 16 general-purpose inputs or outputs at TTL levels. The direction of the input and output lines on this connector are set by jumpers E16 and E17. Further software settings are required to configure this port. See the Software Setup section for details.

Note 11: To configure MO1 - MO8 as inputs install jumper E16. To configure MO1 - MO8 as outputs remove jumper E16.
Note 12: To configure MI1 - MI8 as inputs install jumper E17. To configure MI1 - MI8 as outputs remove jumper E17.
Note 13: Includes a 10K ohm pull-up resistor to +5V.

J10 (JHW) Handwheel Encoder Connector



Pin#	Symbol	Function	Description
1	GND	Common	Reference voltage
2	+5V	Output	Supply voltage
3	HWA1+	Input	HW1 channel A+
4	HWA1-	Input	HW1 channel A-
5	HWB1+	Input	HW1 channel B+
6	HWB1-	Input	HW1 channel B-
7	HWA2+	Input	HW2 channel A+
8	HWA2-	Input	HW2 channel A-
9	HWB2+	Input	HW2 channel B+
10	HWB2-	Input	HW2 channel B-
11	PUL1+	Output	PULSE1+ output
12	PUL1-	Output	PULSE1- output
13	DIR1+	Output	DIRECTION1+ output
14	DIR1-	Output	DIRECTION1- output
15	PUL2+	Output	PULSE2+ output
16	PUL2-	Output	PULSE2- output
17	DIR2+	Output	DIRECTION2+ output
18	DIR2-	Output	DIRECTION2- output
19	TBD		
20	TBD		
21	TBD		
22	TBD		
23	HWANA+	Output	OPT12 Filtered PWM DAC+
24	HWANA-	Output	OPT12 Filtered PWM DAC-
25	GND	Common	Reference voltage
26	+5V	Output	Supply voltage

J12 Ethernet Port

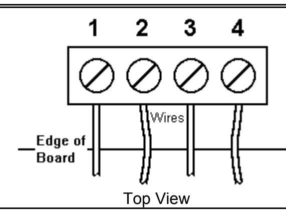
Pin #	Function
1	TXD+
2	TXD-
3	RXD+
4	No Connect
5	No Connect
6	RXD-
7	No Connect
8	No Connect
9	No Connect
10	No Connect

The appropriate Category 5 10/100-Base T network cable that mates to this connector can be readily purchased from any local computer store. The type of network cable to purchase depends on the configuration to the host PC.

When making a direct connection to a Host communication Ethernet card in a PC, a Cat 5 networking crossover cable must be used. A standard Cat 5 straight-through networking cable cannot be used in this scenario. When using a connection to a network hub or switch, the standard Cat 5 straight-through networking cable must be used, and not a crossover cable.

TB1 (JPWR): Power Supply

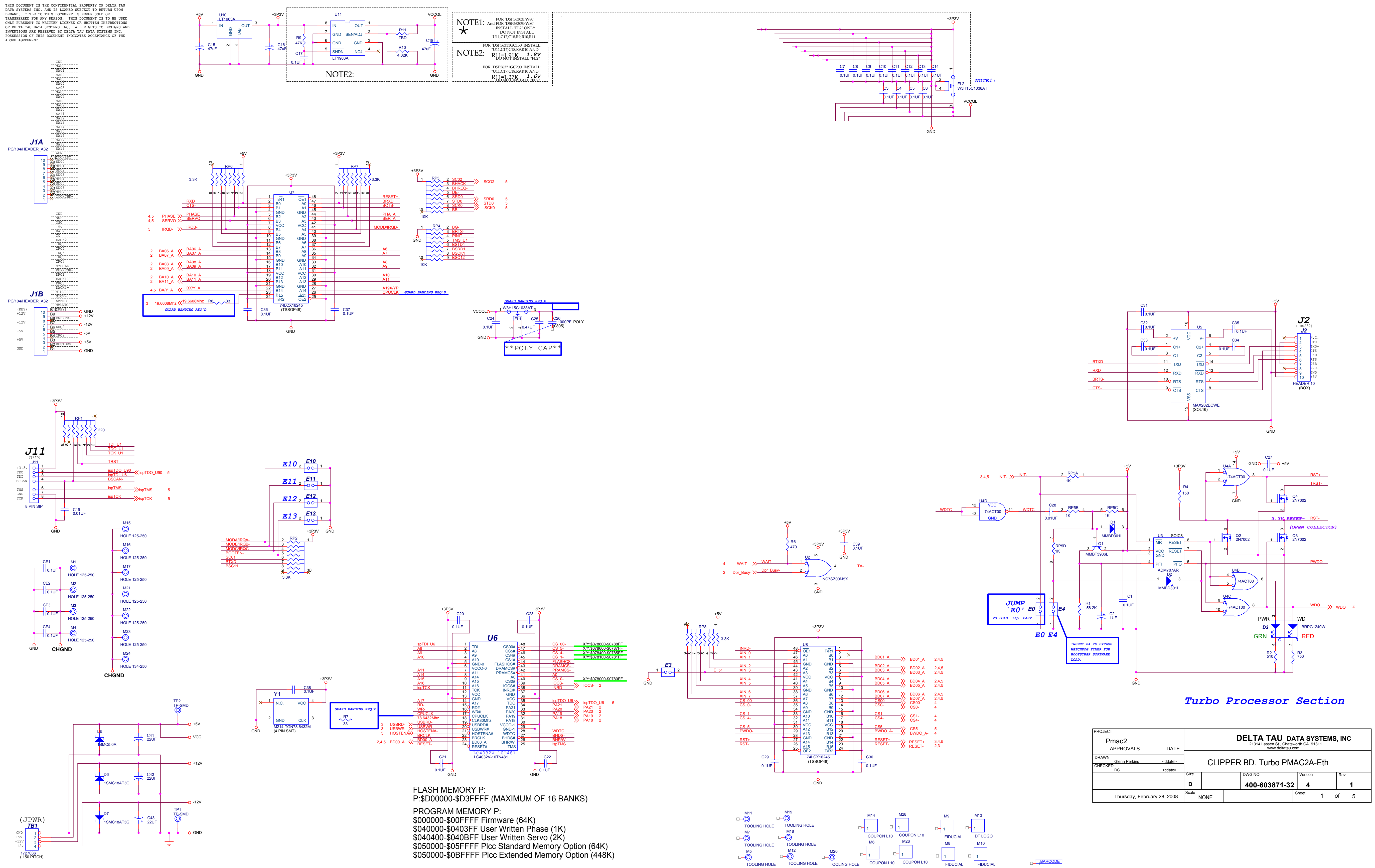
(4-Pin Terminal Block)



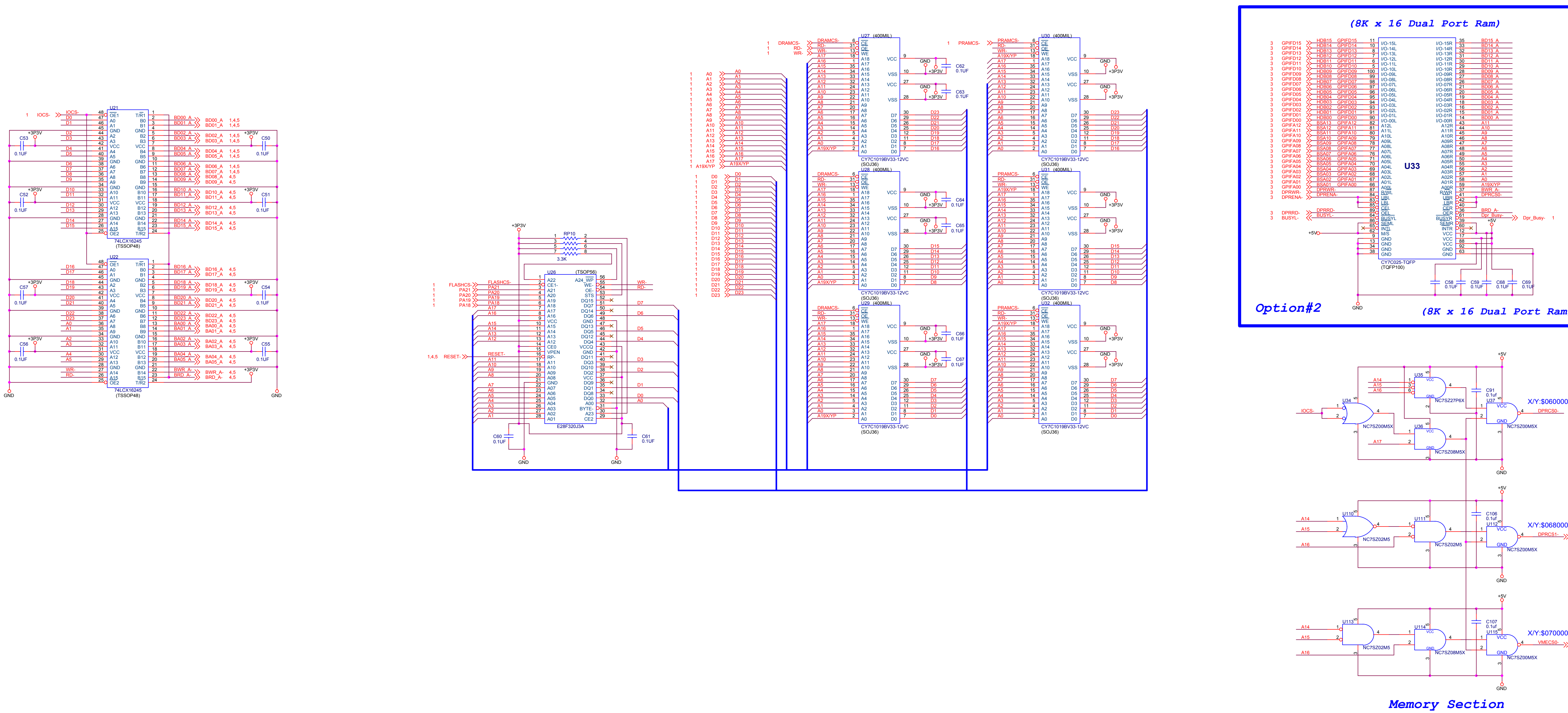
Pin#	Symbol	Function	Description	Notes
1	GND	Common	Digital Common	
2	+5V	Input	Logic Voltage	Supplies all PMAC digital circuits
3	+12V	Input	DAC Supply Voltage	Ref to Digital GND
4	-12V	Input	DAC Supply Voltage	Ref to Digital GND

This terminal block should be used to provide the input for the power supply for the circuits on the PMAC board. For +5V and GND 18 AWG stranded wire is recommended. For +12V and -12V a minimum of 22 AWG stranded wire is recommended.

SCHEMATICS



FLASH MEMORY P:
P:$D00000-$D3FFFF (MAXIMUM OF 16 BANKS)
PROGRAM MEMORY P:
$000000-$00FFFF Firmware (64K)
$040000-$0403FF User Written Phase (1K)
$040400-$040BFF User Written Servo (2K)
$050000-$05FFFF Plcc Standard Memory Option (64K)
$050000-$0BFFFF Plcc Extended Memory Option (448K)
Turbo Processor Section
DELTA TAU DATA SYSTEMS, INC
21314 Lassen St., Chatsworth CA. 91311
www.deltatau.com
PROJECT Pmac2
CLIPPER BD. Turbo PMAC2A-Eth
DRAWN Glenn Perkins
CHECKED DC
Size D
DWG NO 400-603871-32
Version 4
Rev 1
Thursday, February 28, 2008
Scale NONE
Sheet 1 of 5



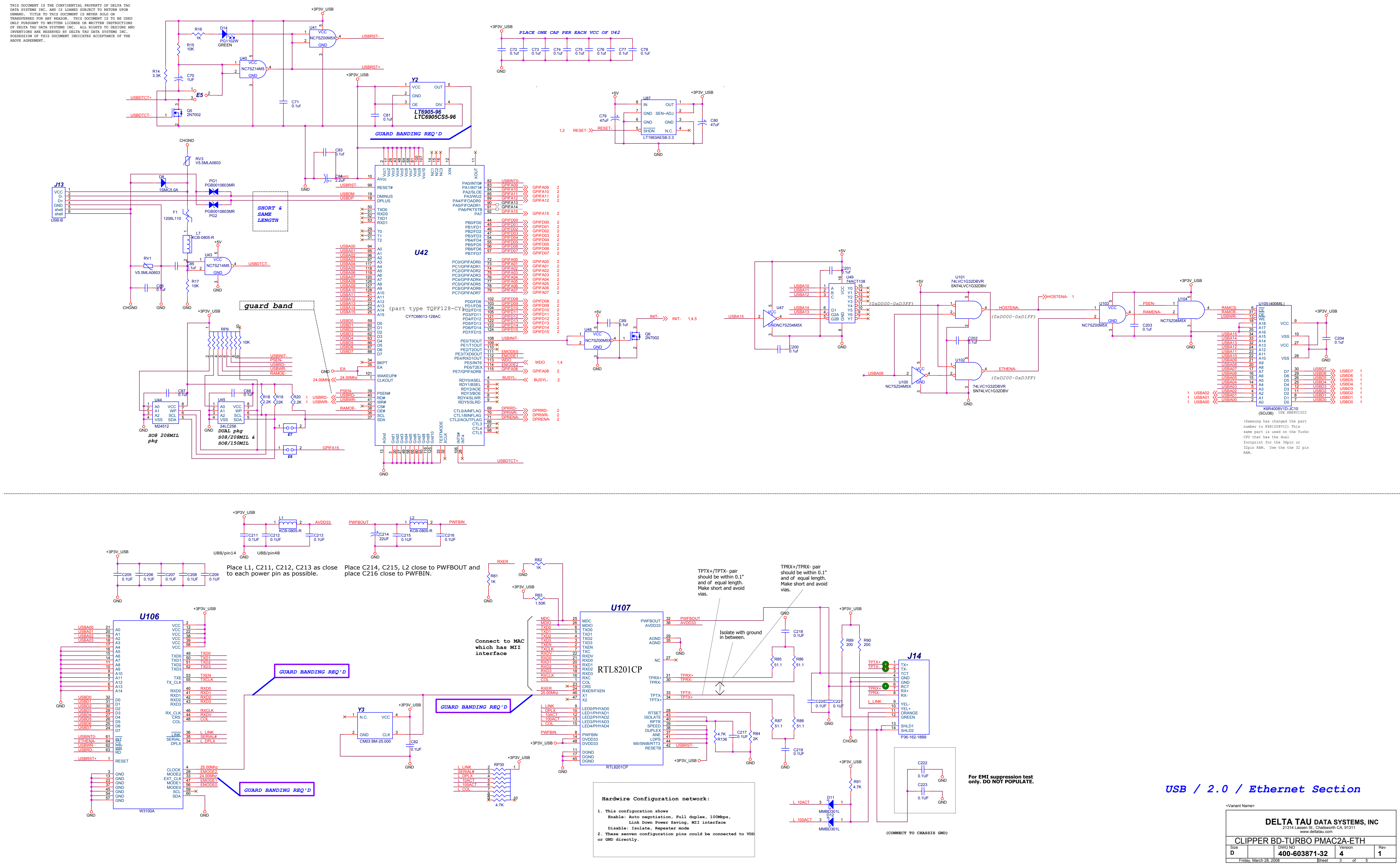
<Variant Name>

DELTA TAU DATA SYSTEMS, INC
21314 Lassen St., Chatsworth CA. 91311
www.deltatau.com

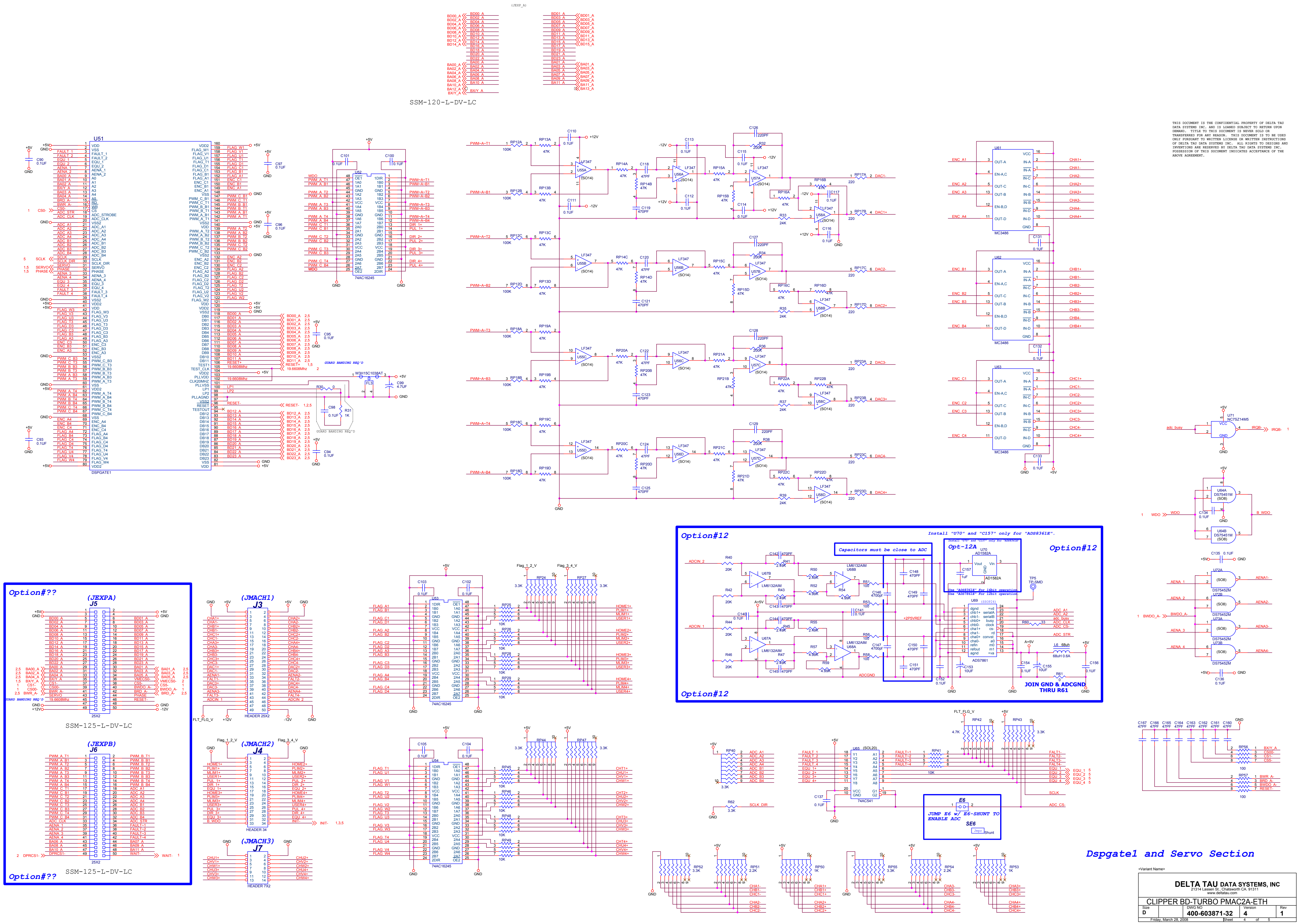
CLIPPER BD-TURBO PMAC2A-ETH

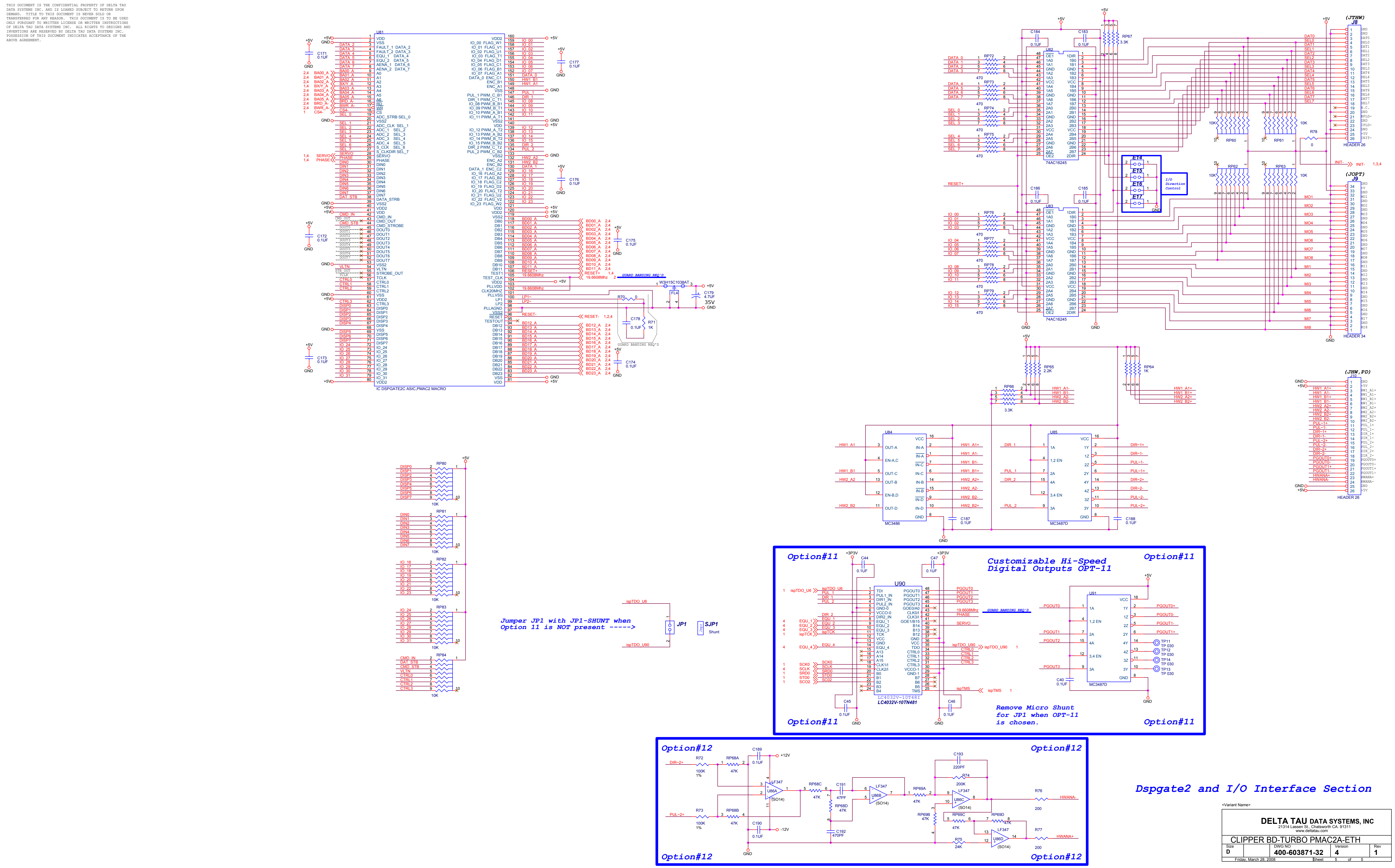
Size		DWG NO	Version	Rev
D		400-603871-32	4	1

Friday, March 28, 2008 — Sheet 2 of 5



GUARD BANDING REQ'D
SHORT & SAME LENGTH
guard band
PLACE ONE CAP PER EACH VCC OF U42
U42
U106
U107
RTL8201CP
J13
J14
Place L1, C211, C212, C213 as close to each power pin as possible.
Place C214, C215, L2 close to PWFBOUT and place C216 close to PWFBIN.
Connect to MAC which has MII interface
Isolate with ground in between.
For EMI suppression test only. DO NOT POPULATE.
(CONNECT TO CHASSIS GND)
Hardwire Configuration network:
USB / 2.0 / Ethernet Section
DELTA TAU DATA SYSTEMS, INC
21314 Lassen St., Chatsworth CA 91311
CLIPPER BD-TURBO PMAC2A-ETH
400-603871-32


SSM-120-L-DV-LC
Option#??
(JEXPA)
J5
SSM-125-L-DV-LC
(JEXPB)
J6
SSM-125-L-DV-LC
Option#??
(JMACH1)
J3
(JMACH2)
J4
(JMACH3)
J7
Option#12
Capacitors must be close to ADC
Opt-12A
Option#12
JOIN GND & ADCGND THRU R61
Option#12
JUMP E6 w/ E6-SHUNT TO ENABLE ADC
Dspgate1 and Servo Section
DELTA TAU DATA SYSTEMS, INC
CLIPPER BD-TURBO PMAC2A-ETH
400-603871-32


Option#11
Customizable Hi-Speed Digital Outputs OPT-11
Jumper JP1 with JP1-SHUNT when Option 11 is NOT present ----->
Remove Micro Shunt for JP1 when OPT-11 is chosen.
Option#12
Dspgate2 and I/O Interface Section
DELTA TAU DATA SYSTEMS, INC
CLIPPER BD-TURBO PMAC2A-ETH
400-603871-32